MODCOMP
MODULAR COMPUTER SYSTEMS, INC

MAX IV System Guide

Distributed Communication Handlers

SOFTWARE

213-804009-100

MODCOMP
MODULAR COMPUTER SYSTEMS, INC.

MAX IV System Guide

Distributed Communication Handlers

SOFTWARE

MANUAL HISTORY

Manual Order Number: 213-804009-I00

Title: MAX IV SYSTEM GUIDE, DISTRIBUTED COMMUNICATION HANDLERS

Software Part Number: 8240I

Revision Level	Date Issued	Description
I00	08/85	Initial Issue (I.1).

Contents subject to change without notice.

PROPRIETARY INFORMATION

THIS MANUAL CONTAINS CONFIDENTIAL PROPRIETARY INFORMATION PROTECTED BY SOFTWARE PROPERTY RIGHTS AND IS MADE AVAILABLE UPON THE CONDITION THAT THE SOFTWARE CONTAINED HEREIN WILL BE HELD IN ABSOLUTE CONFIDENCE AND MAY NOT BE DISCLOSED IN WHOLE OR IN PART TO OTHERS WITHOUT THE PRIOR WRITTEN PERMISSION OF MODULAR COMPUTER SYSTEMS, INC.

FOR GOVERNMENT USE THE FOLLOWING SHALL APPLY:

RESTRICTED RIGHTS LEGEND

USE, DUPLICATION OR DISCLOSURE BY THE GOVERNMENT IS SUBJECT TO RESTRICTIONS AS SET FORTH IN PARAGRAPH (h) (3) (B) OF THE RIGHTS IN TECHNICAL DATA AND COMPUTER SOFTWARE CLAUSE IN DAR7-104.9(a) (OR SUCH EQUIVALENT FPR, NASPR, AND/OR DOEPR CLAUSE AS MAY BE APPLICABLE).

MODULAR COMPUTER SYSTEMS, INC.
1650 WEST McNAB ROAD
FORT LAUDERDALE, FL 33309

Audience
The information in this manual is directed towards the systems programmer who is responsible for the completed DCS system. Material in this manual is also pertinent for the applications programmer for developing device dependent code. The reader should be familiar with MODCOMP communications terminology.

Subject
This manual discusses the theory behind the Distributed Communication Subsystem (DCS). Detailed information about the specific handlers supported by DCS, currently, the Asynchronous and Bitsynchronous handlers, is presented.

Product Supported
MAX IV Software System, Model 8240I Revision I.1 or later.

Related Publications

Manual Order Number	Manual Title
213-804001-REV	MAX IV GENERAL OPERATING SYSTEM System Guide (SGM)
213-804002-REV	MAX IV SYSGEN System Guide (SGM)
213-804003-REV	MAX IV EXECUTIVE (REX) SERVICES System Guide (SGM)
213-804004-REV	MAX IV OPERATOR COMMUNICATIONS System Guide (SGM)
213-804005-REV	MAX IV BASIC INPUT/OUTPUT SYSTEM System Guide (SGM)
213-804006-REV	MAX IV UNIT RECORD DEVICE HANDLERS System Guide (SGM)
213-804007-REV	MAX IV DATA STORAGE DEVICE HANDLERS System Guide (SGM)
213-804008-REV	MAX IV COMMUNICATION HANDLERS System Guide (SGM)

MODCOMP Product Training
MODCOMP's Education Services provides training courses on many hardware and software products at our Training Center in Ft. Lauderdale, Florida. On-site courses at customer sites can also be arranged. For more information about the courses offered by Education Services, contact the MODCOMP Training Registrar.

TABLE OF CONTENTS

CHAPTER 3 DCS ASYNCHRONOUS COMMUNICATIONS HANDLER

CHAPTER 4 DCS SDLC/HDLC FRAME HANDLER

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
DISTRIBUTED COMMUNICATION SUBSYSTEM THEORY OF OPERATION

This chapter presents the functional description of the Distributed Communications Subsystem (DCS). The pieces of software exclusive to DCS are discussed. The individual handlers, the async, bitsync, etc., that correspond to standard MAX IV handlers are discussed fully in their own chapters.

1.1 OVERVIEW OF DCS

The Distributed Communications Subsystem (DCS) is a new generation of communications controllers. It consists of many intelligent Line Interface Modules (LIM) interfaced by one Block Multiplexor Controller (BMX) to a standard I/O bus. Each LIM includes a microprocessor and supports communication ports. Significantly, the handlers for the communications ports do not run in the CLASSIC but on the LIM cards. Therefore the microprocessor on the LIM card functions as a **co-processor** attached to the CLASSIC.

1.1.1 Features of DCS

The design and nature of DCS marks a departure from the philosophy driving communications controllers on the CLASSIC. The DCS interfaces directly to the standard I/O bus. No communications processor, CP, is required. Therefore, multiple DCS subsystems can be configured on a single CLASSIC. The following features characterize DCS:

- Faulty LIM cards can be replaced with the system powered on and operating.
- At the MAX IV REX I/O level, DCS is compatible with the 1907 Async, Bitsync and MAXNET handlers.
- The Bitsync handler supports the X.25 product.
- DCS handlers can run channels at 19.2 bits per second.
- All communications handlers run in the LIM cards, not in CLASSIC memory, thereby reducing MAP 0 requirements.
- Most DCS handlers support an optional line buffer associated with each input channel to capture and prevent the loss of data that is received when no read is queued.
- The Async handler supports type ahead.
- An extended data chaining format is supported to permit the chain and data segments to span multiple task maps.
- Most DCS data structures relative to the communications port reside on the LIM card, not in CLASSIC memory.

1.1.2 Programming Considerations

Although DCS is software compatible with the 1907 at the user I/O level, DCS is not handler level software compatible with the 1907. Due to the packet oriented, block multiplexed nature of DCS software, significant operating differences exist between DCS and all previous communication controllers, and the user is cautioned not to make assumptions about handler level I/O based upon previous experience. Some significant programming considerations follow:

- Data does not "dribble in" to the user's data buffer as it is received. Rather, data is **buffered in LIM memory** and block transferred to the user buffer when the handler detects an end-of-record.

- The user cannot queue a read with a buffer size larger than the expected read and check for end-of-record as the data comes into the buffer.

- The **maximum transfer count** for read or write operations to a DCS subchannel must be specified beforehand at system generation time. The LIM validates and uses this value at download time to allocate data buffers in LIM memory. Subsequent read or write operations cannot exceed this size.

- The **extended data chaining** format is the only data chaining format supported.

- For **full duplex** communications ports, the following parameters must be described identically on both input and output channels:

  Baud Rate
  Quick or Controlled Initiate Mode
  Frame Size
  Stop Bits

- Channel loopback, or internal wrap, is supported by the diagnostics but not by the standard handlers.

- The HOLD bit in the UFT is reset after the end-of-record is detected on the LIM, not as the first character is received into the buffer.

1.2 HARDWARE DEFINITION

Each DCS chassis has 20 slots. The leftmost 4 slots house the Block Multiplexor (BMX) and optionally the Remote Host Option (RHO). The right 16 slots house the Line Interface Modules (LIM). On the back of the chassis and fitting directly onto the backplanes are 16 locations for the Driver/Receiver Modules (DRM). Each DRM is directly opposite to a LIM slot.

LIM/DRM pairs can be fitted into any of the 16 positions available. There is no need for contiguity. LIMs can be fitted with the power on and the system running thereby increasing system up time. DRMs also can be fitted if the LIM is not plugged in. The user connections are cabled directly to the DRMs. A CLASSIC system has a total connectivity potential of 2048 lines as follows:

- Each LIM handles up to 4 communications ports.
- Each DCS takes 16 LIMs.
- Each I/O bus can attach up to 8 DCS chassis.
- Each CLASSIC can attach up to 4 I/O buses.

1.2.1 Block Multiplexor

The Block Multiplexor (BMX) is connected directly to the CLASSIC. Its purpose is to provide arbitration among the LIMs and to interface the CLASSIC I/O bus to the backplane bus of the DCS. Each BMX uses one DMP channel for half-duplex, high speed transfers. The BMX provides command decode, control, interrupt and status returns to the CPU. It arbitrates among the LIMs and performs block data transfers between LIM and CPU.

1.2.2 Line Interface Module

Each Line Interface Module (LIM) is an intelligent interface between the VME bus and the Driver/Receiver Module (DRM). The LIM is microprocessor driven. Each LIM has a M68000 microprocessor, 128KB of RAM, 4 full duplex DMA channels, 4 full duplex universal serial I/O channels and logic to interface to the VME bus. Each LIM can run any combination of asynch, bitsynch or MAXNET lines.

1.2.3 Driver/Receiver Module

The Driver/Receiver Module (DRM) contains the mechanical and electrical interfaces for physically connecting and driving the communications lines. Electrical signals to and from the LIMs are fed to drivers and receivers on the DRM that convert the signals to the levels needed to drive the communications lines.

1.3 SOFTWARE DEFINITION

To optimize the hardware architecture, DCS software is composed of interacting modules that communicate with one another through a software defined packet mechanism in order to transfer user data. DCS software modules reside either in the CLASSIC or in the LIM card. The following software runs in the CLASSIC:

- SYSGEN and SIOGEN Macros
- OC Directives
- SIO Loader Symbiont Task
- Block Mux Handler (BMH)

The software that runs in the LIM cards is as follows:

- LIM Monitor
- Asynchronous Communications Handler
- SDLC/HDLC Frame Handler
- MAXNET IV Link Handler

1.3.1 LIM Software

There are two categories of software that run on the LIM cards: support software and runtime software. Support software consists of diagnostics. Runtime software consists of the Monitor and the individual handlers. Runtime software is defined by the following DCS software modules:

- Monitor
- Asynchronous handler
- Bitsynchronous handler (SDLC/HDLC Frame Handler)
- MAXNET handler

Each of these software modules is unique to the DCS software package. The asynchronous and bitsynchronous handlers correspond functionally to standard MAX IV handlers and are discussed in the following chapters. The MAXNET handler is not discussed in this manual, but can be found in the software publications supporting the MAXNET product.

1.3.2 DCS Disc Partitions

The DCS software requires two disc partitions. One for data structures and one for software that is to be downloaded into the LIM cards. The disc partitions must be called HAN and TBL and the user should verify that there is no conflict with user defined disc partitions. These partitions are defined on the standard MAX IV software system.

HAN contains the binary absolute code of the Monitor and the relocatable binary of the handlers to be downloaded into the LIM cards. **The user should not alter the contents of the HAN partition.**

TBL contains the binary of the data structure to be downloaded into the LIM cards. It is created during the SIOGEN process from the user supplied statements. The data structures, the physical device tables (PDTs) for each communications port, are **not resident in MAP 0.** The **only MAP 0 data structure** is the PDT for each DCS, a task control table (TCT) for each DCS and an logical device table (LDT) for each communication port. The PDTs for each communications port are assembled to the TBL disc partition. The SIO Loader task downloads these data structures into each LIM card.

1.4 THEORY OF OPERATION

The CLASSIC makes high level requests to the LIM and counts on the microprocessor on the LIM to understand and execute the complex requests. To the user, REX I/O requests are unchanged. The I/O requests are queued to the PDT for the Block Multiplexor (MUX) Handler, BMH. Each request is formatted into a request packet and sent to the handler on the appropriate LIM by the Block MUX Handler (BMH). The handler at the LIM performs the operation and sends the results of the operation back to the the BMH on the CLASSIC in a status packet. The entire DCS architecture is optimized for packet transfers.

1.4.1 Block MUX Handler

All I/O operations directed to any DCS communications port are queued to the Block MUX Handler. The BMH moves the nodes from the Block MUX PDT to the MIAP PDT (LIM PDTs read from disc). The Block MUX Handler is responsible for packaging the I/O requests and transmitting them to the LIM cards. The I/O request packet consists of command blocks with commands directed to both the controller and the handlers on the card, and the data block. The Block MUX Handler (BMH) receives, along with data blocks, status packets from the handlers on the LIMs. Through these status packets the following functions are supported:

- The line support task (default is TMP) is activated when informed by the Async Handler of a break detect.
- The attentive task is activated when the LIM sends notification of a Ring event and the Auto Task Line Connect is optioned.
- The attentive task is deestablished and killed when a Ring event occurs and the Auto Task Line Disconnect is optioned.

A watchdog timer is started when the transfer initiate command is issued to start the packet transfer. If this timer expires, the controller is presumed inoperative and is set offline. All outstanding operations for this controller are terminated.

The task level portion of the Block MUX Handler performs standard tasks such as validating the buffer address and incrementing the DPI. If the requested operation is chained, the Block MUX Handler steps through the user's chain list to verify the following:

- The extended mode is used (bit 0 of MIAP is set). If this test fails, the user gets a DCS EXT abort.
- The number of data segments does not exceed ten. Greater than ten segments yields a DCS CHN abort.
- The sum of the transfer counts in each segment does not exceed the maximum transfer size. If this occurs, the user gets a DCS BUF abort. The maximum transfer size is calculated by the Monitor on the LIM based on certain SYSGEN and SIOGEN parameters, and is passed to the Block MUX Handler during the last stage of the download process.

1.4.2 LIM Monitor

The Monitor is the first section of code downloaded to the LIM at startup time. A relocating loader routine is activated which receives the code for up to four different handlers from the SIO Loader task in the CLASSIC and relocates them into the appropriate locations in LIM memory. When loading is complete, the Monitor activates the individual handlers.

The Monitor serves as a simple **taskmaster** by calling each of the handlers in a roundrobin fashion. It is responsible for receiving incoming packets from the CLASSIC and routing them to the appropriate handlers. It also routes packets from the handlers to the CLASSIC. The Monitor provides routines for allocating, queuing, and deallocating data buffers, for processing interrupts using default or handler provided interrupt routines, and for handling the LIM Serial Communications Controller (SCC) and Direct Memory Access (DMA) devices on the LIM. The Monitor also provides timer services to the handlers. A handler can start or stop a timer and is automatically invoked by the taskmaster when a timer expires.

1.5 DOWNLOADING THE LIM CARDS

The SIO Loader task resides on the CLASSIC. It is cataloged as a privileged, nonresident, autostart symbiont task. At system startup time, the SIO Loader reads the LIM PDTs from the disc partition TBL into actual memory. The LIM PDTs were generated by the SIOGEN statements (during SYSGEN). The SIO Loader establishes linkages between the LDTs and the PDTs for each communications port and stores a "slot table address" in the PDT for the Block Multiplexor Handler. The SIO Loader sets up the MIAP PDT for each communications port with the following information:

- LCT address for MAXNET channels.
- TCL address for non-MAXNET channels.
- Copy of LDTSTA, LDTTYP, TCLSTA, TCLDEL, IO$LSC, IO$LRC, IO$LRI and IO$BRC
- LIM Monitor and handlers from the HAN partition
- Copy of a portion of the MIAP PDTs

Once the linkages have been completed, the SIO Loader loads the LIM Monitor, a copy of a portion of the MIAP PDT, and the handlers from the HAN disc partition onto each LIM card. The SIO Loader task queues a REX,WRITE with the UDW bit set in the extended option word of the UFT, to the Block MUX Handler to fill a LIM. Word -1 of the UFT contains the loader option word. Words -2 and -3 contain the starting addresses for the load. When the Block MUX Handler processes this request, a command is issued to the controller indicating that a LIM is to be filled. The load is a privileged operation. Errors during the load are reported to the operator. The SIO Loader has exclusive use of the Block Multiplexor PDT during the downloading sequence. The SIO Loader task exits from the system upon completion.

1.5.1 LIM Downloading Sequence

The following sequence of events typifies a successful downloading process. This process is accomplished only once at system startup. Following the successful completion of the DCS system, all transfers between the LIM and the BMH are accomplished with pre-defined packets.

- Power up LIM card.
- LIM runs a health check then interrupts the controller.
- Boot the MAX IV/DCS system.

- The SIO Loader symbiont reads the LIM PDTs from partition TBL, does the necessary linkages.

- The SIO Loader reads in one block of LIM Monitor at a time from HAN partition, sending each block down to every LIM card before reading the next block. No status packets are sent from the LIM to the CLASSIC during the downloading of the Monitor. The Write UFT for the Monitor block to the LIM has a load option word so that when the first block of the Monitor is written to the card, a bit is set in the option word to tell the BMH to halt the card when the last block of the Monitor is written to the card. A bit is set in the option word to tell the BMH to restart the card after sending this block.

- The Monitor performs its initialization routines and then transfers control to the relocating loader routines.

- The SIO Loader reads the configuration word in the MIAP PDTs to see which handlers need to be downloaded to a specific card and only loads those necessary. The Monitor on each LIM contains a relocating loader to receive each block downloaded and relocate it in the LIM. The load option word tells the Monitor which communications ports this handler services.

- The SIO Loader reads in each handler from the HAN partition one block at a time, downloading the block to each card requiring that handler before reading the next block or the next handler. After receiving each block, the Monitor sends a status packet to the CLASSIC.

- The last block of each handler contains the start execution address which is passed by the relocating loader routines to the taskmaster routines of the Monitor.

- After all the handlers have been downloaded to the LIMs, the SIO Loader downloads the Startup Configuration Parameters (SCP) for each communications port on the card. After the relocating loader routines have finished loading the SCPs, the address of each SCP is given to the Monitor, which sets up a LIM Physical Device Table (PDT) corresponding to each SCP.

- The Monitor calculates a buffer size for LIM data transfers based on certain parameters in the SCPs. The Monitor allocates buffers of this size from the available LIM memory, one per LIM PDT, plus one spare for itself. It then sends this buffer size in a status packet back to the Block MUX Handler on the CLASSIC. The Block MUX Handler uses this buffer size as the maximum transfer size for the LIM and aborts any subsequent REX READ or WRITE requests with a transfer count that exceeds this value.

  Those channels that specify 0 buffers as parameter 8 on the SIOCONTROLLER statement will have their maximum transfer size set to 0 and the exclusive use task name in the LDT set to -1. Any attempt to queue operations to that channel will result in a PXU abort.

- The Monitor invokes its taskmaster routines, allowing each handler to coldstart.

- The SIO Loader task exits.

1.5.2 LCD and DCD Operator Communications Directives

DCS has two operator communications directives which both require the use of the SIO Loader task. For all OC requests, the hold bit (HOL) in the UFT status word is reset when the node is unqueued from the Block MUX PDT and moved to the MIAP PDT queue. Otherwise, HOL is reset when the operation completes. In cases where it is necessary to replace a LIM card, an OC directive is provided to refill one card rather than the whole system. This is the LCD directive and is specified in one of the following formats:

//LCD [group/unit number, slot number] or

//LCD [device name]

This directive activates the SIO Loader task and causes a reload of the LIM Monitor and the appropriate handlers onto the LIM card. The LIM PDTs are not reread from the TBL file. If an error occurs, no retries are attempted. The operator must reissue the LCD directive in the event of an error. The user can also reload all the LIM cards in the DCS system by issuing the LCD directive in the following format:

//LCD ALL

It may be necessary for diagnostic purposes to display the card's memory. This capability is provided through the DCD OC directive in one of the following formats:

//DCD [group/unit number, slot number] or

//DCD [device name]

This directive causes the SIO Loader task to be activated. The SIO Loader issues a REX READ with the UDW bit set when a dump of the memory on the LIM is desired. As in the load, the -1 word of the UFT contains the loader options. Words -2 and -3 contain the starting address of the dump. The Block MUX Handler issues a command to the controller requesting a halt of the LIM and a memory dump. The dump is also a privileged operation. The SIO Loader reads all of the memory from the appropriate card and dumps it to the DO file. The dump is hexadecimal format with the ASCII and CAN-code conversions appended to the right of the dump. If an error occurs, no retries are attempted, and the operator must reissue the DCD directive.

1.6 REX I/O

All REX I/O begins at the user task when the UFT and buffer points are set for the REX call as detailed in Table 1-1. The -3, -2 and -1 words of the UFT are used only for requests from the SIO Loader Task. The user sets up the UFT with the following values for words 0 through 11:

EQUATE	WORD	MEANING
	-3	Monitor Load or dump starting address (MSW) if loader operation (Bit 6 set in XUFOPT, the extendeded option word and Bit 0 set in word -1).
	-2	Monitor load or dump starting address (LSW) if loader operation.
	-1	Load Option Word if UDW (B6) set in XUFOPT extended option word. Bit settings follow: 0 =1, load/dump request 1 =1, first block of Monitor load or dump 2 =1, Monitor load. =0, Handler load. 3 =1, final handler/data structure/Monitor load block.
UFTSTA	0	Status Word 0 ERR =1 and (B4 or B7 set), meaning as described below: 4 MPE =1 extended status words found in UFTBCT and UFTFAT 7 OTH =1 WHO and WHY reason codes in UFTBCT and UFTFAT
UFTNAM	1	File Name
UFTOPT	2	Option Word
UFTDPI	3	Device Position Index
UFTBCT	4	Bytes transferred if operation complete. If ERR=1 and MPE=1 in UFTSTA, UFTBCT defined in Table 1-2.
UFTFAT	5	Logical file index. If ERR=1 and MPE=1 in UFTSTA, UFTFAT defined in Table 1-2.
XUFOPT	6	Extended user's option word. Bit settings are the same as standard with the following exception: 6 UDW =1 indicates that loader information precedes the UFT.
XUFMIA	7	Map Image Actual Page (where bit 0 must be set to indicate extended chaining format) or -2 (extended chaining, operand map) or -3 (extended chaining, instruction map).
XUFBUF	8	Buffer or chain list address.
XUFCNT	9	Byte count or chain map select (-1=MAP 0, -2=(XUFMIA))
XUFADR	10	Double precision position index (MSH)
XUFADL	11	Double precision position index (LSH)

Table 1-1. DCS UFT

1.6.1 Extended Status Words

The user has the option to select user recovery (SR=0 in UFTOPT) mode and perform his own error recovery procedures in the event of an I/O error. To help the user determine the nature of the I/O error, additional status information is returned in words UFTBCT and UFTFAT to the user if an error occurs.

If bits 0 (ERR) and 7 (OTH) of UFTSTA are set, then UFTBCT and UFTFAT contain the "WHO" and "WHY" error reason codes respectively. If system recovery is optioned (SR=1 in UFTOPT), then the "WHO" and "WHY" codes appear on the abort message on the operator's console. In user recovery mode (SR=0 in UFTOPT), the user can interrogate the information in the program.

If bits 0 (ERR) and 4 (MPE) of UFTSTA are set, then UFTBCT and UFTFAT contain extended status words as specified in Table 1-2.

UFTBCT	4	Extended Status Word 'A" bits set as follows:	
		0	set if controller received a parity error
		1	set if controller received a BUF error (locked out from LIM - no buffers available)
		2	set if controller received a memory parity terminate (MPE)
		3	set if Block MUX Handler detected a sequence number mismatch
		4	set if Block MUX controller is inoperable
		5	set if a VME timeout occurred (LIM inoperable)
		7	set if overflow of status packet
		8	set if packet error on LIM
		9	set if DMA error occurred on LIM
		10	set if LIM memory parity error
		11	set if relocating loader detected error
		15	set for any other internal error on card
UFTFAT	5	Extended Status Word 'B" bits set as follows:	
		0	set if CTS (clear to send) dropped
		1	set if CD (carrier detect) dropped
		2	set if DSR (data set ready) dropped
		3	set for line buffer overrun
		4	set if Abort received
		5	set if DAM overflow

Table 1-2. Extended Status Words of UFT

1.6.2 Packet Transfers

Moving user data from the CLASSIC to the LIM is accomplished with packet oriented block transfers from the Block MUX Handler to the LIM handlers. Packet transfers are entirely the responsibility of the DCS software. The user has no interface to packet transfers. There are four types of packets used for operations between the Block MUX Handler and the LIM:

- o Command packet
- o Status packet in response to operation
- o Packet sent in response to data structures load
- o Special events status packet

1.6.3 Write Operation

The following discussion traces the sequence of events for a WRITE operation. The operation starts when a user task sets up the UFT and buffer pointers and does a REX,WRITE. The REX traps into BIOS. BIOS prepares a node and calls the task level portion of the Block MUX Handler (BMH). The task level routine (TL$) calls the BIOS routine IO$IOR, which calls IS$QUE. IS$QUE queues the node to the PDT for the Block MUX Handler and enters the terminate level (TER$) of the Block MUX Handler (BMH) to terminate the controller.

The controller generates a startup SI and control picks up at the SI$ entry point. The BMH moves the node from the Block MUX PDT queue to the MIAP PDT queue. If the channel is not busy, the operation is started. The Block MUX Handler packages the I/O request into a command packet and appends the write data (from the user buffer) to it. It then causes the Block MUX controller to send the packet to the appropriate LIM card.

The controller accesses dedicated locations in the LIM memory and picks up the address of the next available empty packet buffer. It also resets the buffer available flag and transfers an image of the packet into LIM memory. When the transfer is complete, it generates an SI to the Block MUX Handler. The BMH issues a command which causes the controller to set the packet ready flag in LIM memory and generate an interrupt to the LIM.

In response to the interrupt, the LIM Monitor sets the available buffer pointer to the next empty packet buffer and sets the buffer available flag. The Monitor then queues the newly received packet to the addressed communications handler on the LIM. The Taskmaster then causes the service level routine of the handler to run.

The handler is in the next packet state. It recognizes the new packet and enters the startup state. The operation is started and the handler enters the "wait for operation to complete" state. The transfer is now driven by the DMA controller or data interrupts until end of record. At end of record, the handler enters the shutdown state. A "status packet" is prepared and sent to the CLASSIC.

A Monitor routine queues the packet to be sent to the CLASSIC. If no operation is currently pending, it sets a pointer to the packet, sets the incoming packet flag, and generates an interrupt to the controller. The controller reads the first few words of the packet into "quick status" registers and generates an SI to the CLASSIC.

The Block MUX controller reads the controller status and determines this is a packet ready SI. It reads the quick status registers by executing a series of IDx instructions. From this information, it determines this is only the return of status for a WRITE, updates the node and shuts down the operation. The calling task is now resumed and the operation is complete.

1.6.4 Read Operation

The REX,READ operation is similar to the WRITE. As the Block MUX Handler is preparing the command packet, it determines it is a READ and builds only the command part of the command packet since there is no WRITE data to append. The packet is sent and handled by the Monitor at the LIM as in the WRITE case.

During the "wait for operation to complete" state, the data is transferred into LIM memory from the communications line. When end of record is detected, the handler enters the shutdown state. The handler prepares a status packet and appends the READ data to the packet. This packet is sent by the Monitor to the CLASSIC.

During packet ready SI processing the Block MUX Handler identifies the operation and determines that data is expected. It sets up the TA, TC and MAP information and does a transfer initiate command to the controller. The read data is now block transferred directly into the user's data buffer. When the transfer is complete, the controller generates a shutdown SI. The Block MUX Handler then issues a command that causes the controller to set a transfer acknowledge flag in LIM memory and interrupt the LIM.

At this point, the user node is shut down. If other operations are pending, they are started now. Otherwise, interrupts from the LIMs are enabled so that any waiting status packets can be transfered.

CHAPTER 2
SYSTEM GENERATION

This chapter describes the system generation statements unique to DCS software configurations. This information is duplicated in the specific DCS handler chapters along with examples of each handler SYSGEN source.

2.1 SYSTEM GENERATION

There are **two phases** required to generate a DCS system:

- SYSGEN, the system generation phase
- SIOGEN, the communication port PDT generation phase.

SYSGEN requires identifying the Block MUX Controller and the communications port logical devices. SIOGEN is a separate assembly of the communications ports' PDTs. These PDTs are assembled to disc partition TBL from which the SIO Loader task reads them into actual memory at startup time. The assembly of the MIAP PDT, as they are called, can be performed independently of the system generation procedure.

A DCS system generation requires new statements in the I/O block of the standard MAX IV SYSGEN source. The statements in the I/O block generate the data structures that link the DCS with the Basic I/O System (BIOS).

SIOGEN is the independent macro assembly step of the MIAP PDTs. The SIOGEN generates a set of PDT data structures that are placed on the TBL disc partition. These PDTs are brought into actual memory at system startup. The PDTs are not resident in MAP 0. They are accessed through a map image actual page (MIAP) by the Loader task and the Block MUX Handler. A series of slot and subchannel tables also are generated by the SIOGEN. These allow the Block MUX Handler to find the appropriate MIAP PDT when an interrupt occurs.

In addition to the unique DCS SYSGEN statements required, the user is required to specify two vacant LOGFILES in the Task Block for the Operator Communications (OC) Task. These logfiles are used by the SIO Loader Task for downloading code to the LIM cards.

2.2 STANDARD SYSGEN REQUIREMENTS

The SYSGEN statements identified in this section are unique to DCS software. These statements are required in the SYSGEN source. The SIOGEN statements appear in the next section.

2.2.1 MUXCONTROLLER Statement

The MUXCONTROLLER statements must be specified once for each Block MUX Controller in the system. This statement must be specified in the Basic I/O System Block and should appear only after all of the COMCONTROLLER statements. MUXCONTROLLER generates a MAP 0 resident PDT and TCT for the Block MUX Controller. External references are also produced to include the Block MUX Handler during the link edit phase.

SYNTAX

MUXCON[TROLLER]   cname,#gu,dmpl [,wdog] [,link]

cname - is the controller name and must be unique for all controller names

#gu - is the device address. Device addresses are specified as the device group (g) and unit (u) that are unique to the hardware controller. The controller's group and unit number are concatenated into one hexadecimal quantity for this controller.

dmpl - is the number of the Direct Memory Processor (DMP) channel. Depending on the computer configuration, the following numbers are valid:

#00 - #0F for the bus 0
#10 - #1F for the bus 1
#20 - #2F for the bus 2
#30 - #3F for the bus 3

[wdog] - (OPTIONAL) specifies the maximum time in two hundredths of a second that the Block MUX Handler waits for transfer of a message to complete. The default value for this parameter is 50 (1/4 second).

[link] - (OPTIONAL) specifies the total number of MAXNET links configured for this Block MUX Controller.

2.2.2 SIODEVICE Statement

The SIODEVICE statement must be specified for each DCS channel to be defined. This statement generates a Logical Device Table (LDT). The LDTs that are generated are resident in MAP 0.

SYNTAX

```
SIODEV[ICE]      dname,cname[,bpr][,opt][,xtask]
                                     [,-infl]
```

dname - is the logical device name (1 to 3 CAN-code characters) and must be unique among all device names.

cname - is the MUX controller name that was specified as parameter 1 of the associated MUXCONTROLLER statement.

[bpr] - (OPTIONAL) specifies the record size in bytes (bytes per record) for this device. Valid values are 0 through 510 bytes. If 0 (the default) is specified, the record size is considered unlimited. See also the SIOCONTROLLER statement, parameter 7 (MAXTC - maximum transfer count).

[opt] - permits modification of the contents of the logical device characteristics word in the data structure (LDT) for this device. It specifies certain options that may be examined by the specific handler when a handler supports many similar but not identical types of devices. This LDT option word is shown in the specific DCS handler chapters.

[xtask]
[-influ] - (OPTIONAL) specifies either a task name or a negative influence limit. If a task name is given, the designated task is given permanent exclusive use of the device to the complete exclusion of all other tasks.

If a negative influence limit number in the range of -000 or -255 is given, then only tasks that have an influence limit number with an absolute value less than or equal to the absolute value of this parameter are permitted to use this logical device. The default is a negative influence limit number of -255.

2.2.3 Loader Task

Because the SIO Loader Task is to run as a symbiont, the following statements are required in the SYSGEN source:

```
SYMCONTROLLER   SIOLDR
DEVICE          SIO,SIOLDR
PRESCHEDULE     SIOLDR,3,LM
```

2.2.4 OC Task Block

The user must specify two vacant logfile entries in the Task Resource Block of the OC Task. The SIO Loader task dynamically makes assignments with these logfiles during the downloading phase to the LIM cards.

```
LOGFILES        2
```

2.3 SIOGEN REQUIREMENTS

The following statements comprise the SIOGEN. The SIOGEN is a separate macro assembly from the SYSGEN. The data structures generated from the SIOGEN are written to disc partition TBL, dedicated to DCS use.

2.3.1 Insert Macro Prototypes

Before the macro language that defines the PDTs for the communications channels can be used, the prototypes that define the language must be inserted from an external file. This standard macro language statement performs this function and is required in the SIOGEN.

SYNTAX

INS fname,SIOMAC

fname - specifies the logical file name where the macro prototypes are cataloged. The MAX IV standard for this file is MC, the Common Macro Library. This file name must be assigned to a real file or device before it can be used.

SIOMAC - is the name of the macro group used to define the statements in the SIOGEN.

2.3.2 SIOCHARACTERISTICS Statement

The SIOCHARACTERISTIC statement must be specified at least once in the SIOGEN. This statement serves to set all of the defaults for the SIOCONTROLLER statements that follow. This statement can be repeated as many times as necessary in the SIOGEN to change the default parameters.

SYNTAX

SIOCHA[RACTERISTIC] cls,#gu [,csize] [,parity] [,stop] [,echo]
[,delay] [,baud] [,bof] [,sync] [,mnbaud]

cls - specifies the channel class. Enter one of the following keywords to define class of the channel:

A = Asynchronous
B = SDLC/HDLC Frame Level
C = MAXNET Asynchronous
D = MAXNET Bit Synchronous
E = MAXNET Byte Synchronous

#gu - is the device address. Device addresses are specified as the device group (g) and unit (u) that are unique to the hardware controller. The controller's group and unit number are concatenated into one hexadecimal quantity for this controller.

[csize] - (OPTIONAL) specifies the data character size in bits. The range is 5,6,7 or 8. The default is 8. For SDLC/HDLC this parameter must be specified as "DLC" and parameters 4, 5 and 6 take on new meaning as specified in the SDLC/HDLC DCS Handler chapter.

[parity] - (OPTIONAL) specifies the type of parity that is generated/checked. The keywords to select the parity are EVEN, ODD or NONE. The default is NONE. If parameter 3 is 'DLC' this parameter has no meaning.

[stop] - (OPTIONAL) specifies the number of stop bits and has meaning only for asynchronous devices. The values of this parameter can be either 1, 1.5 or 2. The default is 2. If parameter 3 was specified as 'DLC", this parameter has no meaning.

[echo] - (OPTIONAL) specifies the type of echo desired. The keywords used to select the echo type are ECHO or NOECHO. The default is NOECHO. If parameter 3 was specified as 'DLC' this parameter represents the secondary station address in the range of 0 to 255. The default is 0.

[delay] - (OPTIONAL) specifies the delay time (in hundredths of a second) prior to each message being output. This delay time serves the same purpose as the output of NUL characters. The default is 0. This parameter has meaning only for asynchronous devices.

[baud] - (OPTIONAL) specifies the line speed if internal clock is used. Asynchronous lines can support rates up to 19.2K baud. Bit Synchronous lines can support the standard baud rates up to 64K baud. MAXNET supports rates up to 19.2K baud. If external clock is desired, the keyword 'EXT' must be specified.

[bof] - (OPTIONAL) identifies the bottom-of-form simulation support count. This is the number of lines per page. The default is to use the hardware bottom-of-form signal. This parameter has meaning only on output devices or channels.

[sync] - (OPTIONAL) specifies the sync character for synchronous channels. The sync character is specified as a hexadecimal character pair (#16 for ASCII or #32 for EBCDIC). The default is #16.

[mnbaud] - must specify the baud rate for MAXNET links if EXT was specified for parameter 8, baud.

2.3.3 SIOCONTROLLER Statement

The SIOCONTROLLER statement must be specified for each DCS channel to be defined. There must be a one-to-one correspondence between the SIOCONTROLLER statements and the SIODEVICE statements that were specified in the SYSGEN. This statement causes a PDT to be generated. The PDTs that result from the SIOGEN are not resident in MAP 0. Rather, this set of data structures is loaded into a MAP by the SIO Loader task when the system is first started. The SIO Loader also downloads the PDTs for the appropriate channels into the respective LIM cards.

SYNTAX

SIOCON[TROLLER] dname,#chnl [,fdn] [,wdog] [,CL] [,ridmp] [,maxtc] [,ubufs]

dname - is the logical device name of the DCS channel. It must be the same device name as was specified for the corresponding SIODEVICE statement.

chnl - is the subchannel number. It is a hexadecimal value with the upper byte representing the slot number and the lower byte representing the subchannel. For example, #0306 represents slot 3, subchannel 6. Valid slot numbers range from 0 to #F. Valid subchannel numbers range from 0 to 7.

[fdn] - (OPTIONAL) is used only for full duplex channels and specifies the logical device name (dname) of the 'mate'. For simplex channels, this parameter must be left blank.

[wdog] - (OPTIONAL) specifies the maximum time, in hundredths of a second, that an operation is allowed to remain at the head of the queue for this channel. The default value is 0, which means that the operation is not timed.

[CL] - (OPTIONAL) is a keyword used to specify that an asynchronous device has a current loop interface.

[ridmp] - represents the DMP number of the input channel on the satellite if connected to a 4828 on the satellite. This parameter has meaning only for those channels configured for MAXNET.

[maxtc] - (OPTIONAL) specifies the maximum transfer count in bytes for REX READ or WRITE operations on this channel. The default is 2048 bytes. If present, this parameter can be zero only if parameter 8, ubufs, is also zero. The LIM Monitor uses the largest value of this parameter including default values and parameter 3 (bpr - bytes per record) of the SIODEVICE statement to determine the size of the LIM buffers to allocate for data transfers.

[ubufs] - (OPTIONAL) specifies the number of buffers that should be allocated on the LIM for use by the handler for this subchannel. Most handlers need only 1 (the default). The bitsync handler needs a minimum of 3 (its default).

The sum of the values of this parameter for all the subchannels SYSGENed for a LIM, plus one, makes up the total number of identically sized buffers that the LIM Monitor will allocate on the LIM. How the buffer size is determined is described under parameter 7, maxtc.

If too many large buffers are requested and there is not enough memory on the LIM, the user will get an error from the Loader task indicating that there is insufficent memory on the LIM card. To correct this, reduce the value of this parameter or parameter 7, maxtc, or parameter 3, bpr, of the SIODEVICE statement.

2.3.4 ENDSIO Statement

The ENDSIO statement is required. It serves to finalize the data structures that are generated in the SIOGEN. It also terminates the macro assembly by outputting an END statement.

ENDSIO

There are no parameters for this macro.

2.4 SUMMARY OF DCS SYSGEN AND SIOGEN STATEMENTS

The following statements are required in the Logical I/O Block of the SYSGEN source:

MUXCON	DC1,#05,5	one per DCS Block MUX Controller
.		
.		
.		
SYMCOM	SIOLDR	one per system
.		
.		
.		
DEVICE	SIO,SIOLDR	one per system
.		
.		
.		
SIODEVICE statements here		one statement per half-duplex channel, two per full-duplex

The following statement appears in the Task Structure Block of the SYSGEN source:

PRESCHEDULE	SIOLDR,3,SM	one per system
(OC TASK BLOCK)		
LOGFILES	2	in OC Task Block

The following statements occur in the SIOGEN assembly:

INS	MC,SIOMAC
SIOCHARACTERISTICS	statement
SIOCONTROLLER	statements for similar devices
.	
.	
.	
SIOCHARACTERISTICS	statement defining new characteristics
SIOCONTROLLER	statements for similar devices
.	
.	
.	
ENDSIO	

CHAPTER 3
DCS ASYNCHRONOUS COMMUNICATIONS HANDLER

The DCS Asynchronous Communications Handler provides the capability of communicating with local or remote asynchronous data terminals or other computers. The handler consists of several modules that provide the means for a MODCOMP user to effectively read and/or write data to terminals connected to MODCOMP communications channels. This handler resides on the LIM cards in a DCS system. The handler operates with standard and nonstandard ASCII data as well as formatted and unformatted binary data and supports half- and full-duplex operations.

3.1 FEATURES OF THE ASYNCHRONOUS HANDLER

The DCS Asynchronous Handler provides the user with the capability of communicating with local or remote data terminals. It supports the REX service functions and UFT options supported by the standard asynchronus communication handlers AS.HAN and AS7.HAN. To the MAX IV user, the operations performed are functionally identical to the standard handlers. The following features differ from the standard handlers:

- Internal wraparound mode is not supported. An online diagnostic is provided which provides this capability.
- Data is stored in the user's buffer only upon successful completion of a read. This data is block transferred and does not "trickle" into the user buffer. If the read is unsuccessful, no data is returned to the user.
- The HOLD bit (HOL) in the UFT status word is reset when the operation is completed rather than after the first byte has been transferred. If the request is from the Operator Communications Task, HOL is reset when the node is queued to the MIAP PDT.
- Data chaining is supported using the extended chaining format only.
- The delay between form feeds and/or line feeds is accomplished through the use of a delay timer rather than pad characters.
- Stopping output through an XOFF takes several character times.
- A line buffer is associated with each input channel. Input data arriving while the channel is inactive is stored in the buffer. Reads queued to a channel read data from the line buffer first, unless the user specifically requests that the line buffer not be used. This can be accomplished by setting bit 11 in the extended option word of the UFT when the read is queued. Additionally, the line buffer is set to an empty condition following a break event.

3.2 BREAK-DETECT FUNCTION

Break-detect functions are supported by the asynchronous handler if the terminal is SYSGENed as requiring this capability. The break-detect function is supported in two ways by the handler. The first break-function is the break-key, on terminals so equipped.
The second break-function supported is the "break character" (CTRL A). The "break

character" is a user-defined character that performs functionally as a break-key. To change the break character to something other than CTRL A (the default), the system manager simply performs a SYSGEN using the BRKCHARACTER system block statement. If a user types this character on a half- or full-duplex terminal while the input channel is inactive or during a nonstandard/standard ASCII read, the handler will perform the same function as if a break-key were depressed. The "break character" interrupt does not occur during binary read requests.

For a detailed description of the LINEMONITOR and BRKCHARACTER SYSGEN statements refer to the MAX IV SYSGEN, System Guide. The user requiring break detect functions should also refer to Terminal Monitor Program (TMP), a chapter in the MAX IV GENERAL OPERATING SYSTEM, System Guide. TMP monitors break supported terminals.

3.3 OPERATIONS AND MODES

The asynchronous communications capability is based on the most common asynchronous device: a teletype or CRT type device. The handler contains half-duplex and full-duplex operational capabilities.

Unit records passed to the handler are manipulated to conform to the standards specified for a particular type of device. These operations are governed by the data content of the logical device table. The handler does analyze data passed to it so that it might prefix carriage return/line feed operations to the unit being addressed.

When the asynchronous communications channels are created during the system generation procedures, several options are available to describe the type of terminal that is to be attached to a particular communications channel. Parity may be specified in three ways: even parity, odd parity, or no parity. The data character size of the device may be specified as five bits, six bits, seven bits, or eight bits of data. This character size specifies the actual number of bits transferred into or out of memory per character. If seven bits are specified plus parity, the high order (parity) bit is reset so that the data appears as 7-bit ASCII in the calling program's buffer. The number of stop bits may also be specified as one or two depending upon the specific characteristics of the terminal attached to a channel.

A watchdog timer can be connected to each asynchronous channel to provide for all output operations to be timed. The timing factor assigned for each asynchronous device is specified in the SIOGEN.

3.4 STANDARD INPUT/OUTPUT FORMATS

Four basic modes of data transfer are specified by Bits 2 and 3 of the UFT option word (Word 2). These modes are summarized in Table 3-1.

3.5 TERMINAL LISTING CONTROL

Terminal listing control (TLC) allows the terminal user to hold or resume terminal output asynchronously with respect to the listing. This ability is selected at SYSGEN in the option parameter of the SIODEVICE statement.

The SIODEVICE option parameter for either a half-duplex device or the output channel of a full-duplex device can enable this feature for a particular terminal. The user sets bit 6 of

the LDT option word equal to one to enable terminal listing control. Before selecting optional functions, other device characteristics should be considered. System managers should refer to the description of the SIODEVICE SYSGEN statement presented in this chapter.

Terminal listing control is supported for DI and DMA transfers. The channel interface must be full-duplex, however, logically the channel may be configured as either half- or full-duplex.

Bit 2	Bit 3	Mode	
0	0	ASCII Mode	\ Nonstandard modes (for device-dependent programming)
0	1	Binary Mode	/
1	0	ASCII Mode	\ Standard MODCOMP modes (for device-independent programming)
1	1	Binary Mode	/

Table 3-1. Standard I/O Formats

3.5.1 Listing Control Characters

The user controls listing output to a terminal by using the TLC characters. The user enters the listing stop character to stall the listing output to the terminal. The system default listing stop character is CTRL S. The user can continue the listing output to the terminal by entering the listing resume character. The system default listing resume character is CTRL Q.

The system manager can change the listing control characters at SYSGEN. The selected characters are identified on a LISTCONTROL SYSGEN statement.

3.5.2 Automatic Listing Resume

Another parameter optionally specified on the LISTCONTROL SYSGEN statement permits listings that have been held to be resumed automatically following a fixed time interval. The handler resumes output transmission following the specified time interval as if a listing resume character had been received from the associated terminal input channel. Refer to the LISTCONTROL statement in the MAX IV SYSGEN, System Guide.

3.5.3 TLC Functionality

Since a user is able to use terminal listing control at any time during a terminal session, its functionality is explained during both input and output operations.

3.5.3.1 Output Operation Active

When an output operation to the terminal is active:

- If the user enters the listing stop character, the associated terminal system handler stops output to the terminal as soon as the current output operation completes.
- If the listing stop character is entered when an output listing has already been held, no action occurs.
- If the user enters the listing resume character, the associated terminal system handler attempts to resume output to the terminal as quickly as possible.
- If the listing resume character is entered when an output listing is in progress, no change occurs.

3.5.3.2 Output Operation Not Active

When an output is not active:

- If the user enters the listing stop character, the associated terminal system handler inhibits all future output to that terminal until a listing resume character is received.
- If the user enters the listing resume character, the associated terminal system handler allows output to occur to that terminal when output requests are received by the handler.

3.5.3.3 Standard/Nonstandard ASCII Read Active

When a standard/nonstandard ASCII read is active and a TLC character is input, concurrent full-duplex output listings are affected as previously described. The listing control character is not placed in the buffer associated with the current read operation.

3.5.3.4 Standard/Nonstandard Binary Read Active

When a standard/nonstandard Binary read is active and a TLC character is input, no listing control function is performed. The listing control character is placed in the buffer associated with the current read operation.

3.5.3.5 No Input Request Active

When a user input request is not active and a TLC character is input, half- and full-duplex output listings are affected as previously described. The listing control character is discarded.

3.6 BUFFERED DEVICE SUPPORT

Buffered device support (BDS) is available in the MAX IV Asynchronous handler. The user selects buffered device support at SYSGEN by setting a bit in the option parameter of the SIODEVICE statement.

Before selecting optional functions, other device characteristics should be considered. System managers should refer to the description of the SIODEVICE SYSGEN statement presented in this chapter.

The SIODEVICE option parameter for either a half-duplex device or the output channel of a full-duplex device can enable this feature for a particular buffered device. There are two ways to interface support to a buffered device. One involves RS232C signal monitoring and the other relies on input data patterns.

Channels supporting buffered devices that use the RS232C interface technique must be SYSGENed as RS232C channels (not current loop). The user sets bit 1 of the LDT option word equal to one to enable buffered device (RS232 Interface) support for a particular channel.

Logically the channel can be configured as half-duplex or full- duplex. Physically the channel interface must be full-duplex if buffered device X/ON-X/OFF support is desired. Buffered device X/ON-X/OFF support must be configured as current loop. The user sets bit 0 of the LDT option word equal to one to enable buffered device (X/ON-X/OFF) support for a particular channel.

3.6.1 Watchdog Timer

The watchdog timer specifies the maximum time, in hundredths of a second, that an operation is allowed to remain at the head of the queue in the queuing data structure at the LIM card. Buffered device output is timed in this fashion when a watchdog value is specified on the associated SIODEVICE statement. The timer value may need to be increased due to the delay possibilities of buffered devices.

3.6.2 X/ON-X/OFF Support

X/ON-X/OFF character support is handled in a manner very similar to that used by terminal listing control. Buffered devices supported by the Asynchronous Communications Handler send a "Transmission ON" (X/ON) or a "Transmission OFF" (X/OFF) character to the I/O system to signal conditions within the device. The X/OFF character (#13) indicates the device's buffer is full. Receipt of an X/OFF character during output transmission causes the output to be stopped as quickly as possible.

X/OFF characters received when the output channel is non-busy cause the handler to delay further user output until an X/ON character is received.

When the buffer full condition is eliminated the device sends an X/ON character (#11). Receipt of an X/ON character while the output channel is non-busy causes the handler to allow output transmission and to continue any previously delayed listing. X/ON characters received when the output channel is busy outputting cause no change to the handler's action.

3.6.3 RS232C Support

The RS232C signal support technique is primarily controlled from the hardware channel level. Normal "controlled" output transmissions are required to wait for Clear To Send (CTS) and Data Set Ready (DSR) to be enabled before they are started. If during the transmission either DSR or CTS drops, the transfer is terminated with a malfunction error. Malfunction

detection processing is modified for channels connected to buffered devices. Fluctuations in the CTS signal cause the desired stop/go output transmission affect.

The buffered device's handshaking signal line needs to be connected to the channel's CTS signal. When CTS is true, it allows output transmission to proceed. CTS dropping delays output transmission. These channels must be SYSGENed as RS232C interfaces (not current loop). Modems cannot be utilized on channels where buffered device (RS232C Signal) support is optioned. (Therefore, buffered devices using this method cannot be used on the same line with ring detect support.)

3.6.4 X/ON - X/OFF Character Handling

Since buffered terminals are supported, it is necessary to specify the handling of the receipt of an X/ON or X/OFF character.

- When a standard/nonstandard ASCII read is active, concurrent full-duplex I/O is affected. X/ON and X/OFF characters are not placed in the buffer associated with the current read operation.

- When a standard/nonstandard binary read is active, no change occurs to output transmission. X/ON and X/OFF characters are placed in the buffer associated with the current read operation.

- When a user read is not active, half- and full-duplex output transmissions are affected. The X/ON and X/OFF characters are discarded.

3.7 I/O OPERATIONS AND MODES

The handler's response to each I/O operation is described in the following sections. The effects of the basic modes of data transfer are also presented.

3.7.1 Standard ASCII Read

On standard ASCII read operations each character can be returned to the terminal (ECHO) by setting the ECHO option indicator in the logical device table. This option is selected by parameter 4 of the SIODEVICE SYSGEN statement. The ECHO feature is not supported for all devices.

On a read operation, the unit record transmitted is a line of up to 72 characters and is usually delineated by a special ASCII byte called the terminating character. Carriage widths from 4 to 510 bytes may be specified as a unit record limit for a device at system generation. For the standard ASCII mode, the terminating character can be either a carriage return (CR) or a trailing NUL (gap) character. Leading NUL characters are discarded unless the DDO bit is set in the UFT option word. Leading characters are control characters received before a valid data character is received by the handler. Trailing characters are control characters received after at least one valid data character. The control character line-feed (LF) is "filtered out" on standard ASCII operations and is never put into the user's data buffer whether leading, trailing, or intermediate.

3.7.1.1 Prefix Message

Before a read operation is initiated to a keyboard, a short prefix consisting of a carriage return and a line feed is transmitted to the terminal. This prefix serves to:

- Inform the operator that an input is queued.
- Position the carriage on a unit record boundary (start of next line) in case it was left in an indeterminate position by the previous print or input operation.

This carriage positioning prefix can be defeated by the device-dependent option bit in the UFT if the user wishes to proceed with input on the same line as a previous print operation. Of course, the program must deal with an interactive teletypewriter, in such a case, rather than with the general input device that standard ASCII mode attempts to simulate.

3.7.1.2 Special Purpose Characters

Error correction of operator type-in is local to the handler in this mode and requires that certain ASCII control characters be reserved for special purposes:

The backspace (CTRL H) character is not passed as a data byte but causes the previously entered data byte and the backspace (BS) itself to be deleted from the input record. If successive BS bytes are entered, they delete an equal number of previously typed characters. If more BS bytes are entered than there are previous characters, the read operation is terminated and restarted without any knowledge of the program.

The delete (DEL) byte (sometimes called RUBOUT) cancels all previously entered bytes in the input record and restarts the read operation, also without program assistance. Thus the typist, not the program, must be concerned with this type of error.

All trailing blanks are suppressed and a null word (two NUL bytes on a word boundary) marks the end of useful information in the core buffer as long as there is still room in the buffer to store these extra bytes. A user can define such a character sequence (DFC 0) at the end of the input buffer if the possibility exists that the input record will fill the entire buffer. This trailing blank suppression feature, though rarely necessary when the keyboard is being operated interactively, ensures that input records received from different devices will look the same to a program.

After a read operation is terminated, the first word (Bytes 0 and 1) is examined by the handler. If it contains the ASCII characters "$$", the end-of-file (EOF) indicator is turned on in the program's UFT.

A special control bit modifies the device handler's response at the end of a unit record. This bit (bit 6 of UFT option word), when set, defeats end-of-file detection, trailing blank suppression, and the storing of the NUL byte marker at the end of useful information. In this special mode, the program can determine the exact number of bytes entered and the end of useful information in the buffer only by a byte count stored in the UFT. This mode can be used in cases where the size of the unit record is only one byte.

A character sequence ....LF/CR....... is treated as a "null" or "blank" record and a NUL marker is put in word 0 of the program's data buffer.

3.7.2 Nonstandard ASCII Read

A read operation in the nonstandard ASCII mode is not as "automatic" as the standard ASCII mode. It requires that the user be aware that the input file is assigned to a specific type of device (that is, teletype keyboard/printer). The main purpose of the mode is to allow for device-dependent programming when the automatic editing and control character interpretation of the standard ASCII mode are restrictive to the user's needs. This mode requires more work on the user's part, however. Such a program cannot have its file indiscriminately assigned to any input device.

This mode retains the automatic CR/LF prefix output to the printer before a read operation is started, but this can be defeated as it could with the standard ASCII mode. None of the local editing bytes (BS or DEL) are interpreted nor are trailing blanks suppressed or line feeds ignored. The size of variable-length unit records may be determined from the byte count returned to the UFT or by the position of a particular terminating character in the buffer.

Leading NUL characters are discarded unless the DDO bit is set in the UFT option word. The user may select a terminating byte when utilizing this mode by placing it in the UFT option word in the correct format. If the user selects not to specify a terminating byte, a leading (if not discarded) or trailing NUL character terminates the input record.

The end-of-file event is never detected in this mode, even if the first two data bytes of the unit record are '$$'.

3.7.3 Standard ASCII Write

The teletypewriter printer transmits one unit record (a line) for each write operation. In the standard ASCII mode, the text transmitted by the program must have a carriage control byte as the first byte (byte 0) in the buffer. This byte is not printed and indicates to the handler which printer carriage positioning function is to be performed before the remainder of the buffer is printed.

Since carriage control is performed before print-out, and as an integral part of the write operation, the operation frees the program from remembering from one line to the next where the carriage is positioned. The carriage control character also facilitates transmission of ultimate carriage positioning information when the printed data is diverted to secondary storage by the system (spooling) without the program's knowledge. A special task (symbiont) can use the carriage control information when it subsequently prints the diverted print data on a physical printer.

The next byte (byte 1) following the carriage control byte can be the start of printed text. However, if this is inconvenient, text can start on the next word (byte 2) boundary if byte 1 is a start-of-text (STX) byte. This deferred text start is often convenient where write operations consist of data recently received by a read command from another device. Since the read operation always stores data in memory starting on a word boundary, the carriage control byte and STX character can simply be put in a word that precedes the read buffer when it is necessary to print. The print and input buffers are thus coincident (though out of phase by one word) and moving of the data is not necessary when the carriage control information is added.

```
PRBUF    DFC #2002      Print carriage control
INBUF    RES 41,0       Input buffer
```

The carriage control bytes are defined in Appendix C of the MAX IV Unit Record Device Handlers, System Guide, however, the Asynchronous handler does not support hardware carriage control.

During a standard ASCII write, text bytes are transmitted until a null (NUL) byte marker is found in the user's buffer or until the device's record size or the buffer's specified size is exceeded. If a NUL byte is used to terminate the output text, the NUL character is not transmitted, however, it is counted in the byte transfer count. Thus, a facility for both variable and fixed- length record sizes is provided, specified either by call argument or by the contents of the data buffer. If the user's buffer is larger than the device's record size, the additional characters are not printed. The use of the NUL byte marker is convenient as a terminator when the program generating the messages is not related to the program calling the write operation. The latter can always be made to specify a buffer that is the maximum device size while the former can communicate to the handler the size of the useful text line by imbedded control bytes in the buffer.

Except for the conventions described above, all other text bytes and control bytes are transmitted to the printer. Even additional carriage control characters may be imbedded in text (e.g., CR, LF, FF) but this convention should not be used if the program might need to have its file assigned to other printing devices that may not control carriage motion with data bytes (e.g., line printer). The use of the carriage control bytes and other imbedded non-printing bytes in the buffer must be accounted for in the device record size and buffer size. All printing devices generally have their record sizes specified two bytes larger than actual carriage width at system generation to account for carriage control bytes. A 72-column printer has 74 bytes specified; a 132 column printer has 134 bytes specified; and so on.

Nonprinting terminals such as CRT's generally have their record size as the actual screen width or two characters less. Some CRT's force an automatic carriage return and line feed when the end of line is reached; therefore, if the user outputs a line feed as a part of the carriage control a double line feed will result. The user should know the characteristics of the terminal in use in order to determine this value.

3.7.4 Nonstandard ASCII Write

In the nonstandard ASCII mode, no carriage control character is interpreted on write operations; thus, all carriage control must be performed by imbedded control bytes in the data text buffer. Therefore, the program must know that the file is assigned to AOn before using this mode.

This mode transmits all characters in the program's data buffer until one of the following occurs:

- The device record size is exceeded
- The specified buffer size is exceeded, or
- A byte in the buffer matches the byte stored in the program's UFT option word.

Thus, variable-length records, whose size is determined by imbedded data bytes, can be transmitted in this mode also. If bit 8 of the option word is set, however, the feature is defeated and only fixed-length records can be transmitted. This mode normally terminates upon finding a NUL byte in the data buffer if the program takes no action to set the desired byte in the UFT or to defeat the feature. The terminating byte is not transmitted, however, it is counted in the byte transfer count.

3.7.5 Binary Data

Nonsymbolic data is transmitted bit-for-bit from memory to device. If the target device is byte oriented, one 8-bit data byte in memory is transmitted to/from the 8-bit wide data path of the device. Two such bytes reside in each 16-bit memory location of the program's data buffer. This record format requires that bytes 0, 2, and 3 be reserved for control purposes so that such records can be identified as binary records and so that variable-length binary records can be defined by control bytes within the unit record itself. This handler has a standard binary and a nonstandard binary mode if either format is restrictive to the program's application.

3.7.6 Binary Read

The binary modes must be used when full 8-bit data bytes are to be transmitted from the asynchronous device. The standard binary mode requires a strict format for the first four bytes of the record. These four data bytes uniquely identify the type of record (standard binary) and its total length in bytes. The nonstandard binary mode has no strict format but unit record sizes must be known to the program since only fixed block-length records (the size of the specified data buffer) can be read. If the user wishes to read 8-bit data the user must do so using nonstandard binary.

Both modes will skip leader (NUL bytes = '00000000) on each unit record, a feature that can be defeated if byte 0 of a nonstandard binary record might contain a NUL byte. In the case where NUL bytes might start a record, it is suggested that the first record (dummy, if necessary) not be allowed to begin with a NUL byte. This will allow the leader-skipping function to operate. The program can defeat the leader-skipping feature after this first record is read. All subsequent records could then contain a NUL byte as the first byte of a record. Without such a convention, the device must be accurately positioned to data byte 0 of the first record. The leader-skipping function can be defeated by setting the DDO bit in the UFT option word.

If a file mark record (an ASCII record) is encountered in the standard binary mode, the EOF event bit is set in the program's UFT. This mechanism can be defeated by setting bit 6 in the UFT option word. If the size of a standard binary record is larger than the program's buffer, the portion of the record that cannot be contained in the buffer will be skipped. This allows for record scanning where keys are being sought that reside near the beginning of a record and the entire record need not be transmitted into memory.

3.7.7 Binary Write

This mode transmits 8-bit binary data bytes to the device. The standard binary mode requires that the data in the program's data buffer conforms to the standard binary record whose size in bytes must be specified in bytes 2 and 3 of the data record. Byte 0 in the program's data buffer must be a binary record indicator byte ('00000011 or '00000111). Attempting to transmit a record that does not conform to these conventions is considered an error condition and has an associated error reason code of "IOB".

The nonstandard binary mode allows the program complete freedom as to the contents of the data buffer being transmitted. Only fixed block-length records can be specified and transmission will continue until either the byte count or logical record size is exceeded.

3.7.8 Advance Record

The advance record operation causes the printer carriage to be up-spaced regardless of data mode. A carriage return (CR) and a line feed (LF) byte are transmitted. No function is performed for the keyboard for this operation.

3.7.9 Advance File

The advance file operation outputs a carriage return followed by four line feeds regardless of data mode. No function is performed for the keyboard for this operation.

3.7.10 Rewind

The rewind operation outputs a form feed to the terminal regardless of data mode. The programmer is encouraged to perform this operation at the beginning of a program if the position of the carriage is unknown. This can also be accomplished by printing the first line at top-of-form. If a "spooled" device that buffers printing data to secondary storage is used, the first line must be printed at top-of-form. Printers without form-feed hardware simulate form feed by transmission of one carriage return (CR) byte followed by four line feed (LF) bytes. Bit 4 in word 3 of the logical device table must be set at system generation if form-feed capability exists in the device. This operation initializes the file position index to zero and sets EOF and BOM in the UFT status word.

3.7.11 Backspace Record/Backspace File

These backspace operations do not transmit data to the terminal but act as null operations. The EOF and BOM event bits are set in the program's UFT.

3.7.12 Write End-of-File Mark

This write operation causes a "$$" message to be printed with a carriage return (CR) and two line feeds (LF) preceding it. The EOF event bit is set in the program's UFT.

3.7.13 Home - Line Disconnect

A line disconnection can be forced by issuing a REX Home service to the half/full-duplex channel. The request can be issued to either channel when a full-duplex line is being disconnected. The option is defined by the device dependent bits (8 through 15) of the extended option word (XUFOPT) of the UFT as follows:

Bit 12 - DCL - Disconnect line

The function performed by the handler will be the same as if the handler had detected an actual line disconnection. (Refer to the RING DETECT SUPPORT section of this chapter for more information.)

3.7.14 Summary Tables

Asynchronous Handler summary tables appear at the end of this chapter. Table 3-3 contains information about the handler's response to I/O operations in four modes of data transfer. Table 3-4 lists the error and event bits in the UFT status word (UFTSTA). Conditions causing the error or event are also listed.

3.8 RING DETECT SUPPORT

Ring detect support provides a useful interface to auto-answer modems. When ring is detected, data terminal ready is enabled if not enabled already. Break support will be initialized on asynchronous lines where optioned. Finally, the outside world is notified of the event by enqueuing a packet (containing line connect event information) to TMP (or other line monitoring task). If a line disconnection occurs, data terminal ready and break are disabled.

3.8.1 Line Disconnection

A line disconnection sequence is executed for the following conditions:

- An output channel hardware malfunction occurs during a write operation. (NOTE: Buffered device support handles this differently.)
- An input channel hardware malfunction occurs during a read operation.
- DSR drops while either the input or output channel are inactive.

The above conditions are expected to handle most line disconnections. Due to the unique nature of select modems, a small percentage of disconnections will go unnoticed. However, an intelligent task may be able to determine that a line has been disconnected due to its unique communication protocol, for example, a LOGOFF in a TSX environment. In these cases, the task may issue a software requested line disconnection (refer to the REX Home service).

3.8.2 Autotask Line Connect/Disconnect

Auto task line connect support (LINEMONITOR option CONN) and auto task line disconnect support (LINEMONITOR option DISC) cause the system supplied line monitor task (TMP) to perform one additional function (other than the required functions to setup support) as defined below:

- Auto task line connect support causes TMP to establish and activate the current attentive task.
- Auto task line disconnect support causes TMP to deestablish and kill the current attentive task.

Both of these options require that ring support (RING) also be optioned.

3.8.3 Line Disconnect Delay

Table 3-2 shows how ring detect support allows the user to tailor the communications environment to suit his/her unique needs and how the user gains control over each ring event that occurs. (Refer to the TMP Chapter for information concerning the use of the U$COMM User Hook in the MAX IV General Operating System, System Guide.)

LINE CONDITION EVENT	LINE MONITOR OPTIONS			ACTION
	ON	OFF	DO NOT CARE	
RING	RING	CONN	BREAK DISC	1. Call User Hook Routine (U$COMM) 2. Continue/Bypass normal function
RING	RING CONN		BREAK DISC	1. Call User Hook Routine (S$COMM) 2. Activate the attentive task 3. Continue/Bypass normal function
RING		RING CONN	BREAK DISC	No Action
HANG	RING	DISC	BREAK CONN	1. Call User Hook Routine (U$COMM) 2. Continue/Bypass normal function
HANG	RING DISC		BREAK CONN	1. Call User Hook Routine (U$COMM) 2. Deestablish/Kill attentive task 3. Continue/Bypass normal function
BREAK	BREAK		RING DISC CONN	1. Call User Hook Routine (U$COMM) 2. Continue/Bypass normal function

Table 3-2. Control Ring Events

3.9 SYSGEN AND SIOGEN STATEMENTS FOR COMMUNICATIONS CHANNELS

The SYSGEN statements in this section and the philosophy of DCS communications system generation are also described in Chapter 2 of this manual.

The channels on each LIM card are numbered from 0 to 7. All even-numbered channels are the output channels of the pair; the odd-numbered channels are the input channels. Each SIOCONTROLLER macro call references those physical channels that it is defined to service. For half-duplex channels only the even numbered (output) channel of the pair is specified but both channels will actually be used. A SIODEVICE macro call is required for each logical device attached to those channels. Several logical devices can be defined for a physical channel. To change channel characteristics, the SIOCHARACTERISTIC statement must precede the SIODEVICE statement.

3.9.1 SIOCHARACTERISTICS Statement

The SIOCHARACTERISTIC statement must be specified at least once in the SIOGEN. This statement serves to set all of the defaults for the SIOCONTROLLER statements that follow. This statement can be repeated as many times as necessary in the SIOGEN to change the default parameters.

SYNTAX

```
SIOCHA[RACTERISTIC]     cls,#gu [,csize] [,parity] [,stop] [,echo]
                        [,delay] [,baud] [,bof] [,sync] [,mnbaud]
```

cls - specifies the channel class. Enter one of the following keywords to define class of the channel:

A = Asynchronous
B = SDLC/HDLC Frame Level
C = MAXNET Asynchronous
D = MAXNET Bit Synchronous
E = MAXNET Byte Synchronous

#gu - is the device address. Device addresses are specified as the device group (g) and unit (u) that are unique to the hardware controller. The controller's group and unit number are concatenated into one hexadecimal quantity for this controller.

csize - (OPTIONAL) specifies the data character size in bits. The range is 5,6,7 or 8. The default is 8. For SDLC/HDLC this parameter must be specified as "DLC" and parameters 4, 5 and 6 take on new meaning as specified in the SDLC/HDLC DCS Handler chapter.

parity - (OPTIONAL) specifies the type of parity that is generated/checked. The keywords to select the parity are EVEN, ODD or NONE. The default is NONE. If parameter 3 is 'DLC' this parameter has no meaning.

stop - (OPTIONAL) specifies the number of stop bits and has meaning only for asynchronous devices. The values of this parameter can be either 1, 1.5 or 2. The default is 2. If parameter 3 was specified as 'DLC", this parameter has no meaning.

echo - (OPTIONAL) specifies the type of echo desired. The keywords used to select the echo type are ECHO or NOECHO. The default is NOECHO. If parameter 3 was specified as 'DLC' this parameter represents the secondary station address in the range of 0 to 255. The default is 0.

delay - (OPTIONAL) specifies the delay time (in hundredths of a second) prior to each message being output. This delay time serves the same purpose as the output of NUL characters. The default is 0. This parameter has meaning only for asynchronous devices.

baud - (OPTIONAL) specifies the line speed if internal clock is selected. Asynchronous can support rates up to 19.2 Kbaud. Bit Synchronous lines can support the standard baud rates up to 64K baud. MAXNET supports rates up to 19.2K baud. If external clock is desired, the keyword 'EXT' must be specified.

bof - (OPTIONAL) identifies the bottom-of-form simulation support count. This is the number of lines per page. The default is to use the hardware bottom-of-form signal. This parameter has meaning only on output devices or channels.

sync - (OPTIONAL) specifies the sync character for synchronous channels. The sync character is specified as a hexadecimal character pair (#16 for ASCII or #32 for EBCDIC). The default is #16.

mnbaud - must specify the baud rate for MAXNET links if EXT was specified for parameter 8, baud.

3.9.2 SIOCONTROLLER Statement

DEFINE EACH COMMUNICATIONS CONTROLLER CHANNEL

The SIOCONTROLLER statement must be specified for each communications line to be used in the DCS IV system. One physical device table is generated for each statement specified. The PDTs generated from the SIOCONTROLLER statements reside on disc until system startup. At that time, they are loaded into actual memory and are called MIAP PDTs. The SIOGEN is an independent macro assembly from SYSGEN.

SYNTAX

SIOCON[TROLLER] dname,#chnl [,fdn] [,wdog] [,CL] [,ridmp] [,maxtc] [ubufs]

dname - is the logical device name of the DCS channel. It must be the same device name as that specified for the corresponding SIODEVICE statement.

chnl - is the channel number. It is a hexadecimal value with the upper byte representing the slot number and the lower byte representing the subchannel. For example, #0306 represents slot 3, subchannel 6. Valid slot numbers are from 0 to #F. Valid channel numbers range from 0 to 7.

[fdn] - (OPTIONAL) is used only for full duplex channels and specifies the logical device name (dname) of the 'mate'. For simplex channels, this parameter must be left blank.

[wdog] - (OPTIONAL) specifies the maximum time, in hundredths of a second, that an operation is allowed to remain at the head of the queue for this channel. The default value is 0, which means that the operation is not timed.

[CL] - (OPTIONAL) is a keyword used to specify that an asynchronous device has a current loop interface.

[ridmp] - is used only for MAXNET channels.

[maxtc] - (OPTIONAL) specifies maximum transfer count in bytes for REX I/O on this channel. The default is 2048 bytes. If present, this parameter can be zero only if parameter 8, ubufs is also zero. The LIM Monitor uses the largest value of this parameter including default values and parameter 3 (bpr) of the SIODEVICE statement to determine the size of the LIM buffers to allocate for data transfers.

[ubufs] - (OPTIONAL) specifies the number of buffers that should be allocated on the LIM for use by the handler for this subchannel. Most handlers need only 1 (the default). The bitsync handler needs a minimum of 3 (its default).

The sum of the values of this parameter for all the subchannels SYSGENed for a LIM, plus one, makes up the total number of identically sized buffers that the LIM Monitor will allocate on the LIM. How the buffer size is determined is described under parameter 7, maxtc.

If too many large buffers are requested and there is not enough memory on the LIM, the user will get an error from the Loader task indicating that there is insufficent memory on the LIM card. To correct this, reduce the value of this parameter or parameter 7, maxtc, or parameter 3, bpr, of the SIODEVICE statement.

3.9.3 MUXCONTROLLER Statement

DEFINE THE BLOCK MUX CONTROLLER

The MUXCONTROLLER statements must be specified once for each Block MUX Controller in the system. This statement must be specified in the Basic I/O System Block and should appear only after all of the COMCONTROLLER statements. MUXCONTROLLER generates a MAP 0 resident PDT and TCT for the Block MUX Controller. External references are also produced to include the Block MUX Handler during the link edit phase.

SYNTAX

MUXCON[TROLLER] cname,#gu,dmpl [,wdog] [,link]

cname - is the controller name and must be unique for all controller names

#gu - is the device address. Device addresses are specified as the device group (g) and unit (u) that are unique to the hardware controller. The controller's group and unit number are concatenated into one hexadecimal quantity for this controller.

dmpl - is the number of the Direct Memory Processor (DMP) channel. Depending on the computer configuration, the following numbers are valid:

#00 - #0F for the bus 0
#10 - #1F for the bus 1
#20 - #2F for the bus 2
#30 - #3F for the bus 3

[wdog] - (OPTIONAL) specifies the maximum time in two hundredths of a second that the Block MUX Handler waits for transfer of a message to complete. The default value for this parameter is 50 (1/4 second).

[link] - (OPTIONAL) specifies the total number of MAXNET links configured for this Block MUX Controller.

3.9.4 SIODEVICES Statement

DEFINE EACH COMMUNICATIONS "DEVICE" IN SYSTEM'S LOGICAL DEVICE LIST

One SIODEVICE statement (SYSGEN) is required for each SIOCONTROLLER statement (SIOGEN) specified in your MAX IV system. The expansion of this macro creates a logical device table entry (LDT).

The asynchronous communications handlers are designed to function with several different types of devices. However, some of these devices have peculiar characteristics that must be distinguished by the handlers. Bits are set in the LDT option word (LDTTYP) at SYSGEN time to indicate to the handler the device peculiarities. Refer to the "options" parameter 4 and Figure 3-1 below.

SYNTAX

```
SIODEV[ICE]       dname,cname [,bpr] [,opt] [,xtask]
                                            [,-infl]
```

dname - is the logical device name and must be unique to all device names specified (communications channels).

cname - is the MUX controller name which was specified as parameter 1 of the associated MUXCONTROLLER.

[bpr] - is the maximum bytes per record that this device accepts (0-256). If zero is specified, the record size is considered "unlimited".

[opt] - permits modification of the contents of the logical device characteristics word in the data structure (LDT) for this device. It specifies certain options that may be examined by the specific handler when a handler supports many similar but not identical types of devices. This LDT option word is shown in Figure 3-1. The meaning of the bits are explained below. (For MAXNET options see the MAXNET Reference Manual).

[xtask]
[-influ] - (OPTIONAL) specifies either a task name or a negative influence lim t. If a task name is given, the designated task is given permanent exclusive use of the device to the complete exclusion of all other tasks.

If a negative influence limit number in the range of -000 or -255 is given, then only tasks that have an influence limit number with absolute value less than or equal to the absolute value of this parameter are permitted to use this logical device. The default is a negative influence limit number of -255.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
BDX	BDM	N36	N08	CFF	NEK	TLC	PLC			PTR			TRM		

BIT	NAME	MEANING IF SET
0	BDX	Enable Buffered Device X/ON-X/OFF Support.
1	BDM	Enable Buffered Device Modem Signal Support.
2	N36	Output 36 Null characters after LF's. Output 209 Null characters after FF's.
3	N08	Output 8 Null characters after LF's. Output 53 Null characters after FF's.
4	CFF	Device has hardware form-feed capability.
5	NEK	Perform software (handler) echo of input.
6	TLC	Enable Terminal Listing Control.
7	PLC	Device used for Programmable Logic Controller (PLC) Interface
10	PTR	Device capable of printing.
13	TRM	Device has keyboard.

Figure 3-1. Asynchronous Device LDT Option Word

3.9.5 BRKCHARACTER Statement

DEFINE SYSTEM BREAK CHARACTER

The BRKCHARACTER statement is optional. It specifies the system break character used on the main console and remote terminals.

Break character processing is described within the BREAK-DETECT FUNCTION section of this chapter. The system break character defaults to CONTROL "A" if this statement is not included in the SYSGEN source file.

SYNTAX

BRKCHA[RACTER] [character]

character - (OPTIONAL) specifies the system break character, usually expressed as a hex value. It defaults to CTRL A. The system break character should not conflict with listing control characters or buffered device X/ON-X/OFF characters.

3.9.6 LISTCONTROL Statement

DEFINE TERMINAL LISTING CONTROL INFORMATION

The LISTCONTROL statement is optional. It defines the system listing control characters and automatic listing resume time interval. Terminal listing control processing is described in this chapter. The system listing stop character defaults to CTRL S and the system listing resume character defaults to CTRL Q if this statement is not included in the SYSGEN source file.

SYNTAX

LISTCO[NTROL] [lsc][,lrc][,lri]

lsc - (OPTIONAL) specifies the listing stop character. This parameter is usually expressed as a hex value. The default for this parameter is #13 or CTRL S.

lrc - (OPTIONAL) specifies the listing resume character. This parameter is usually expressed as a hex value. The default for this parameter is #11 or CTRL Q.

lri - (OPTIONAL) specifies the automatic listing resume time interval. This parameter specifies the number of ticks in the interval to be used by the handler in automatically resuming a held output listing. The timer operates at the low clock level and begins whenever a stop listing condition is detected upon starting up an output operation. The default for this parameter is 0 which indicates that listings are not to be automatically resumed.

The following characters are RESERVED and may not be used as listing stop/resume characters:

System Break Character (the default is CTRL A (#01))
DELETE/RUBOUT (#7F)
Carriage Return (#0D)
Backspace (#08)
NUL (#00)

3.9.7 LINEMONITOR Statement

DEFINE COMMUNICATION LINE MONITORING CHARACTERISTICS

The LINEMONITOR statement is optional and is used only if a break/ring processing task is to be included in the system. This statement is not required for the Operator Communications Task and its terminal to detect break. This statement is required for each communications line for which break/ring will be queued to some line monitoring support task. One such task is supplied by MODCOMP to provide for remote operator communications and auto answer support (this task is called the Terminal Monitor Program - TMP); however, any number of user tasks may also exist in the system.

SYNTAX

LINEMON output [,input] [,options] [,task] [,prefix] [,delay]

output - is the logical device name on which break, CONTROL "A", ring, and line disconnect are to be detected. If the device is a full-duplex device, "output" is the name of the device on the output channel.

[input] - is only required if the device is a full-duplex device. If the device is half-duplex, either this parameter must not be entered or must be entered the same as parameter 1. If the device on which break, CTRL A, ring, and line disconnect is to be detected is a full-duplex device, "input" is the logical device name of the device connected to the input channel.

[options] - (OPTIONAL) specifies line monitoring characteristics as follows:

BREAK	(#2000)	Break Support
RING	(#1000)	Ring Support
CONN	(#0800)	Auto Task Line Connect Support
DISC	(#0400)	Auto Task Line Disconnect Support
DELAY	(#0200)	Line Disconnect Delay Support

Auto task line connect support, auto task line disconnect support, and line disconnect delay support require that ring support be optioned. The default option is break support.

[task] - (OPTIONAL) specifies the task name that is to be notified (by a packet queued to it) when a break, ring, or line disconnect is detected from the supported device. All these task names must be only 3 characters long. The default task name is TMP (Terminal Monitor Program).

[prefix] - (OPTIONAL) specifies a 3-character CAN-codable task name prefix to be used as the prefix to the initial attentive task name. The 3-character suffix to the task name will be the line number in decimal ASCII. A line number is based on the LINEMONITOR statement ordering. The line number of the first LINEMONITOR statement is "001". The line number of each additional statement is "002", "003", et cetera. If this parameter is not supplied then there will be no initial attentive task name.

[delay] - (OPTIONAL) specifies the number of ticks to delay after detecting a line disconnection. The time delay operates at low clock level. This parameter is used in conjunction with line disconnect delay support (DELAY) only and the default delay is 3 seconds or 600 ticks.

3.9.8 Summary

Define a two DCS controllers for addresses #05 and #06.

```
MUXCONTROLLER     DC1,#05,5
MUXCON            DC2,#06,6
```

Define asynchronous half-duplex subchannels (LDT):

```
SIODEV            A10,DC1,82,#224
SIODEV            A12,DC1,82,#224
SIODEV            A14,DC1,82,#224
SIODEV            A16,DC1,82,#224
SIODEV            A20,DC1,82,#224
SIODEV            A22,DC1,82,#224
SIODEV            A30,DC2,82,#224
SIODEV            A32,DC2,82,#224
SIODEV            A34,DC2,82,#224
SIODEV            A36,DC2,82,#224
```

```
BRKCHARACTER      #07
```

Specifies CTRL G as the system break character.

```
LISTCONTROL       #06,#07
```

Specifies CTRL F as listing stop character and CTRL G as listing resume character.

NOTE: These BRKCHA and LISTCO examples cannot appear in the same system. CTRL G (#07) cannot be defined as both a break character and a listing control character.

```
LISTCO   ,,500
```

Another example of LISTCO. Only one LISTCO statement should be used in the SYSGEN source file. This example specifies an automatic listing resume interval as 500 ticks using the default listing control characters.

Specify corresponding LIMEMONITOR statements for SIODEVICE statements as follows:

```
LINEMONITOR       A10,A10,BREAK,TMP,BAT
LINEMONITOR       A12,A12,BREAK,TMP,BAT
LINEMONITOR       A14,A14,BREAK,TMP,BAT
LINEMONITOR       A16,A16,BREAK,TMP,BAT
LINEMONITOR       A20,A20,BREAK,TMP,BAT
LINEMONITOR       A22,A22,BREAK,TMP,BAT
```

```
LINEMONITOR        A30,A30,BREAK,TMP,BAT
LINEMONITOR        A32,A32,BREAK,TMP,BAT
LINEMONITOR        A34,A34,BREAK,TMP,BAT
LINEMONITOR        A36,A36,BREAK,TMP,BAT

LINEMO             A04,A05,RING,TMP,BAT
```

Specifies full-duplex device with ring support. The attentive task name is BAT001.

```
LINEMO             A06,,BREAK+RING+CONN+DISC+DELAY,TMP,BAT,600
```

Specifies all supported options and a disconnect delay of 600 ticks.

* SIOGEN that corresponds to the SYSGEN above

```
TTL       *** SIOGEN FOR DCS.SY ***
INS       MC,SIOMAC
SIOCHA    A,#05,8,NONE,1,ECHO,0,9600,0        ASYNC DEVICES
SIOCON    A10,#0F00,,0,CL
SIOCON    A12,#0F02,,0,CL
SIOCON    A14,#0F04,,0,CL
SIOCON    A16,#0F06,,0,CL

SIOCHA    A,#05,8,NONE,1,ECHO,0,9600,0        ASYNC DEVICES
SIOCON    A20,#0E00,,0,CL
SIOCON    A22,#0E02,,0,CL

SIOCHA    A,#06,8,NONE,1,ECHO,0,9600,0        ASYNC DEVICES
SIOCON    A30,#0500,,0,CL
SIOCON    A32,#0502,,0,CL
SIOCON    A34,#0504,,0,CL
SIOCON    A36,#0506,,0,CL
```

	ASCII MODES		BINARY MODES	
	Standard	**Nonstandard**	**Standard**	**Nonstandard**
0 READ	Input one record line from keyboard. Output CR or NUL. Output prefix (CR, LF) before input.	Input one record from keyboard (user-specified terminator). No prefix to printer.	Input one record whose length is based on the user-defined byte count contained in the record.	Input one record whose length is based on the user-defined byte count contained in the REX call.
1 WRITE	Output one line to printer. Output prefix as per carriage control byte interpretation.	Output one line to printer. No prefix to printer.	Output one record whose length is based on the user-defined byte count contained in the REX call.	Same
2 REWIND	Output simulated (CR,4LF) or true form feed (FF) to printer. Sets EOF and BOM in UFT status word.	Same	Same	Same
3 BKFILE	Null function sets EOF and BOM events in UFT status word.	Same	Same	Same
4 BKRECORD	Null function sets EOF and BOM events in UFT status word.	Same	Same	Same
5 AVRECORD	Advance carriage of printer one line (CR,LF)	Same	Same	Same
6 AVFILE	Advance carriage of printer four lines (CR,4LF)	Same	Same	Same
7 WEOF	Output file-mark ($$) record to printer with CR,2LF) leader prefixed. Sets EOF in UFT status word.	Same	Same	Same
8 HOME	Enabling and disabling of internal channel wraparound mode and/or line disconnection requests.	Same	Same	Same

Table 3-3. Asynchronous Communications Operations and Modes

Programmer's Error/Event Bit In UFTSTA	Operator's Reason Code	Reason For Error If Set
1	OVF	- Input channel overrun
2	PAR	- Input channel parity
3	INO	- Device does not exist or does not respond - Device inoperable - Watchdog timer has expired - DMI not present
4	MPE	- Extended status in UFTBCT, UFTFAT
5	LOK	- Modem malfunction (DSR, Carrier Detect, or Clear to Send dropped
6	CIG	- Output channel ring detect - Input channel frame error
7	OTH	
8	SBV	
9	EOM End-Of-Medium	- None
10	EOF End-Of-File	- $$CR (for terminal device). - Also set after WEOF, Rewind, or Null operation.
11	BOM Beginning-Of-Medium	- Set after Rewind or Null operation.

Table 3-4. Asynchronous Communications Errors and Events

CHAPTER 4
DCS SDLC/HDLC FRAME HANDLER

SDLC (Synchronous Data Link Control) protocol was introduced by IBM in 1969. HDLC, or (High-Level Data Link Control) is a standard proposed by the International Standardization Organization (ISO). Both protocols are bit-oriented and have similar frame formats. The only major differences between the two is the HDLC capability to have extended address and control fields.

ADCCP (Advanced Data Communications Control Procedure) is a protocol standard proposed by ANSI and it is equivalent to the HDLC protocol; therefore, these channels will also support ADCCP.

The DCS SDLC/HDLC frame handler runs on a LIM card in a DCS system. The handler's function is to perform Input/Output data transfers on behalf of a calling task according to the rules of protocol for SDLC/HDLC. It is designed to be a frame-level handler, and it relies upon the calling task to provide protocol logic. The SDLC/HDLC frame handler works in DMA mode. The SDLC/HDLC frame level handler supports half- and full-duplex modes of operation.

4.1 FRAME DEFINITION

This handler provides a MAX IV user with a frame-level interface to support SDLC/HDLC protocol. A frame is defined as a sequence of contiguous bits, bracketed by beginning and ending flag sequences. A valid frame is at least 32 bits in length and contains an address field, a control field and a frame check sequence. A frame may or may not include an information field. A flag sequence is a unique sequence of 8 bits ('01111110') employed to delimit the beginning and ending of a frame. A typical SDLC frame is shown in Figure 4-1.

FIELD NAME:	FLAG	ADDRESS	CONTROL	INFORMATION	FRAME CHECK SEQUENCE	FLAG
ABBREVIATION:	F	A	C	I	FCS	F
BIT LENGTH:	8	8	8	variable	16	8

Figure 4-1. Typical SDLC Frame

The information that is passed from the user task to the handler and vice versa is contained in the A, C, and I fields.

4.2 OPERATIONS AND MODES

This handler supports seven REX functions:

- Rewind
- Home
- Read
- Write
- Terminate
- Take
- Give

All REX functions support quick-return and wait mode options. The standard MAX IV I/O call and return conventions apply.

The handler makes no distinction between standard ASCII, nonstandard ASCII, standard binary, and nonstandard binary. The handler processes byte strings without checking for any special characters.

When the SDLC/HDLC channels are specified at SYSGEN time, several options are available that control line characteristics. The user must specify transmitter and receiver (information field) length of 8 bits per character. A secondary address can be specified but is not enabled until a REX Rewind (#2) is issued requesting a load secondary address.

4.2.1 Rewind

The Rewind REX service is used by the SDLC/HDLC frame handler to initialize an SDLC/HDLC channel. The control information options are selected by setting bits in the extended UFT option word XUFOPT.

No data is transferred by a rewind request; only mode options are selected. The normal status return from a rewind is both end-of-file (EOF) and beginning-of-medium (BOM) bits set in UFTSTA. Rewind requests should be issued to both the input channel and the output channel.

SYSGEN allows the user to specify only the receiver length, transmitter length, and secondary address. When the system is initially loaded, the following is true:

STATION IS PRIMARY
NON-EXTENDED ADDRESS MODE
NON-EXTENDED CONTROL MODE
NON-GLOBAL
IDLE IN MARK

The user can change these conditions by setting option bits in the extended UFT option word and issuing a REX Rewind to both the input channel and the output channel. The option bits are shown in Figure 4-2.

Dynamic control of the Request to Send (RTS) modem signal is supported. The user can have RTS toggle between output transmissions or request that RTS remain enabled until further notice. The initial system startup state is defined at SYSGEN. Bit 7 in UFT word XUFOPT is used to convey the users dynamic request as follows:

B7 set - RTS is to remain enabled after the next output request until further notice.

B7 reset - The RTS signal is to be enabled at the startup of an output request and disabled at the completion of the same request.

The default case is defined as having RTS toggle between output requests so that system software revision up-grades do not alter the previous operating state. Request to Send dynamic control mode requests must be issued to the output channel and can be issued at any time whether other mode options are specified or not. Request to Send dynamic control is interrogated and programmed by the handler on each receipt of a rewind request and as such users must take this into consideration when preparing UFTs associated with rewind requests.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
							RTS	FLG	RSD	XAD	XCL	PRI	SEC	GBL	LSA

BIT	NAME	MEANING WITH A REWIND
7	RTS	= 0 Enable RTS at startup of output request, disable at completion of the same request = 1 Keep RTS enabled after next output request
8	FLG	= 0 Causes idle to occur in mark mode. = 1 Causes idle to occur in flag mode.
9	RSD	= 1 Reset the data terminal ready signal.
10	XAD	= 1 Select the extended address mode.
11	XCL	= 1 Select the extended control mode.
12	PRI	= 1 Select the primary station.
13	SEC	= 1 Select the secondary station.
14	GBL	= 1 Select the global addressing mode.
15	LSA	= 1 To load the secondary address from the configuration word of MIAP PDT.

Figure 4-2. Extended UFT Option Word (XUFOPT)

4.2.2 Home

Associated with each input channel is a read ahead buffer. Reads coming into this channel, while there is no read operation queued from the host, go into this buffer. The size and number of buffers to be used by each channel can be selected in the SIOCONTROLLER statement. This feature is enabled at handler cold start.

The read ahead buffers can be normalized, or flushed, by issuing a REX,HOME with bit 10 of the XUFOPT set. The REX,HOME should be issued to the channel before the task starts to queue reads to the channel.

Bit 10 - Normalize read ahead Buffers

A line disconnection can be forced by issuing a REX Home service to the half/full-duplex channel. The request can be issued to either channel when a full-duplex line is being disconnected. The option is defined by the device dependent bits (8 through 15) of the extended option word XUFOPT of the UFT.

Bit 12 - DCL - Disconnect Line

The function performed by the handler is the same as if the handler had detected an actual line disconnection except that line disconnect delay support, if optioned, is not performed.

4.2.3 Read

A start-of-buffer address and a byte count are required as per standard MAX IV I/O calls. System error recovery is optionally provided. All read requests (including those that encounter errors) are returned to the user if system error recovery is not specified. When the read is marked completed, that is, when the buffer busy (BB) bit and UFT busy (UB) bits are reset, UFTBCT will contain the total number of bytes read.

If the logical device to which the read is directed is in primary mode, extended address, or extended control, the next frame, properly bound by flags, is passed to the user's read buffer. If the logical device is a secondary station, not extended address and not extended control, then the receive will only be completed by frames with the correct secondary address. If the secondary station is a global secondary, it will also receive frames with the global address #FF. Refer to Figure 4-3. The SDLC/HDLC handler, at the completion interrupt of a read, opens the next read on the queue, if present, to receive consecutive frames. There is a small amount of setup time necessary to accomplish this and, therefore, at higher baud rates the receiver may not be enabled in time.

4.2.4 Write

A start-of-buffer address and a transfer byte count are required as per standard MAX IV I/O calls. System error recovery is optionally provided. All write requests, including those that encounter errors, are returned to the user if system error recovery is not specified. When the write is marked complete (BB & UB reset), UFTBCT contains the total number of bytes written. A single frame is written for each write issued.

The function of prematurely terminating a data link is called 'transmit abort'. This function is reserved to the transmitter under SDLC/HDLC procedures. The transmitting station aborts by sending eight consecutive binary ones.

Conditions for transmit abort issued by the hardware are UNDERRUN ON THE TRANSMITTER. Condition when transmit abort is issued by the software (HANDLER):

LIM MEMORY PARITY ERROR ON THE TRANSMITTER
REX TERMINATE OF THE ACTIVE NODE ON TRANSMITTER

RECEIVER STATION MODES:	FRAME ADDRESS FIELD CONTENTS:		
	= Station Address	≠ Station Address	= #FF Global Address
Primary	R	R	R
Secondary Non-extended Address and Control	R	I	I
Global Secondary Non-extended Address and Control	R	I	R
Extended Address or Extended Control (Primary or Secondary)	R	R	R

where: R = Received
I = Ignored

Figure 4-3. READ Request Response

4.2.5 Terminate

A REX Terminate of an active node on a transmitting channel results in a transmit abort being issued by the transmitting channel. A REX Terminate of an active node on a receiving channel results in the receiver operation being terminated.

4.2.6 Take, Give

The handler supports take exclusive use and give exclusive use according to standard procedures, although these operations are handled by the Block MUX Handler rather than on the LIM card.

4.3 RING DETECT SUPPORT

Ring detect support provides a useful interface to auto-answer modems. When ring is detected, Data Terminal Ready is enabled if not enabled already. The outside world is notified of the event by queuing a packet (containing line connect event information) to TMP (or other line monitoring task). If a line disconnection occurs, Data Terminal Ready is disabled.

4.4 TIMER SERVICES

In order to implement watchdog timers for I/O requests using this handler, the user must do the following:

- Specify the watchdog parameter in the SYSGEN SIOCONTROLLER statement.

- If a timer value other than the default is required, the user may supply that value in hundredths of a second in register 3 when the REX call is made. The user must also set the UST bit in the UFT extended option word XUFOPT. When UST is equal to 1, it indicates that the timer value is specified in register 3. When the UST bit reset (equal 0), the SYSGENed value is used for the timer.

4.5 SYSTEM GENERATION AND SIOGEN

The philosophy of SYSGEN and SIOGEN is discussed fully in Chapter 2. The physical channels on a LIM card are numbered from 0 to 7. All even-numbered channels are the output channels of the pair; the odd-numbered channels are the input channels. Each SIOCONTROLLER macro call references those physical channels that it is defined to service. A SIODEVICE macro call is required for each logical device attached to those channels. Several logical devices may be defined for a physical channel.

4.5.1 MUXCONTROLLER Statement

The MUXCONTROLLER statements must be specified once for each Block MUX Controller in the system. This statement must be specified in the Basic I/O System Block and should appear only after all of the COMCONTROLLER statements. MUXCONTROLLER generates a MAP 0 resident PDT and TCT for the Block MUX Controller. External references are also produced to include the Block MUX Handler during the link edit phase.

SYNTAX

MUXCON[TROLLER] cname,#gu,dmpl [,wdog] [,link]

cname - is the controller name and must be unique for all controller names

#gu - is the device address. Device addresses are specified as the device group (g) and unit (u) that are unique to the hardware controller. The controller's group and unit number are concatenated into one hexadecimal quantity for this controller.

dmpl - is the number of the Direct Memory Processor (DMP) channel. Depending on the computer configuration, the following numbers are valid:

#00 - #0F for the bus 0
#10 - #1F for the bus 1
#20 - #2F for the bus 2
#30 - #3F for the bus 3

[wdog] - (OPTIONAL) specifies the maximum time in two hundredths of a second that the Block MUX Handler waits for transfer of a message to complete. The default value for this parameter is 50 (1/4 second).

[link] - (OPTIONAL) specifies the total number of MAXNET links configured for this Block MUX Controller.

4.5.2 SIOCHARACTERISTIC Statement

The SIOCHARACTERISTIC statement must be specified at least once in the SIOGEN. This statement serves to set all of the defaults for the SIOCONTROLLER statements that follow. This statement can be repeated as many times as necessary in the SIOGEN to change the default parameters.

SYNTAX

```
SIOCHA[RACTERISTIC]   cls,#gu [,csize] [,parity] [,stop] [,echo]

                      [,delay] [,baud] [,bof] [,sync] [,mnbaud]
```

cls - specifies the channel class. Enter one of the following keywords to define class of the channel:

A = Asynchronous
B = SDLC/HDLC Frame Level
C = MAXNET Asynchronous
D = MAXNET Bit Synchronous
E = MAXNET Byte Synchronous

#gu - is the device address. Device addresses are specified as the device group (g) and unit (u) that are unique to the hardware controller. The controller's group and unit number are concatenated into one hexadecimal quantity for this controller.

[csize] - (OPTIONAL) specifies the data character size in bits. The range is 5,6,7 or 8. The default is 8. For SDLC/HDLC this parameter must be specified as "DLC" and parameters 4, 5 and 6 take on new meaning as specified in the SDLC/HDLC DCS Handler chapter.

[parity] - (OPTIONAL) specifies the type of parity that is generated/checked. The keywords to select the parity are EVEN, ODD or NONE. The default is NONE. If parameter 3 is 'DLC' this parameter has no meaning.

[stop] - (OPTIONAL) specifies the number of stop bits and has meaning only for asynchronous devices. The values of this parameter can be either 1, 1.5 or 2. The default is 2. If parameter 3 was specified as 'DLC", this parameter has no meaning.

[echo] - (OPTIONAL) specifies the type of echo desired. The keywords used to select the echo type are ECHO or NOECHO. The default is NOECHO. If parameter 3 was specified as 'DLC' this parameter represents the secondary station address in the range of 0 to 255. The default is 0.

[delay] - (OPTIONAL) specifies the delay time (in hundredths of a second) prior to each message being output. This delay time serves the same purpose as the output of NUL characters. The default is 0. This parameter has meaning only for asynchronous devices.

[baud] - (OPTIONAL) specifies the line speed if internal clock. Asynchronous can support rate up to 19.2K baud. Bit Synchronous lines can support the standard baud rates up to 64K baud. MAXNET supports rates up to 19.2K baud. If external clock is desired, the keyword 'EXT' must be specified.

[bof] - (OPTIONAL) identifies the bottom-of-form simulation support count. This is the number of lines per page. The default is to use the hardware bottom-of-form signal. This parameter has meaning only on output devices or channels.

[sync] - (OPTIONAL) specifies the sync character for synchronous channels. The sync character is specified as a hexadecimal character pair (#16 for ASCII or #32 for EBCDIC). The default is #16.

[mnbaud] - must specify the baud rate for MAXNET links if EXT was specified for parameter 8, baud.

4.5.3 SIOCONTROLLER Statement

DEFINE EACH COMMUNICATIONS CONTROLLER CHANNEL

The SIOCONTROLLER statement must be specified for each communications line to be used in the DCS IV system. One physical device table is generated for each statement specified. The PDTs generated from the SIOCONTROLLER statements reside on disc until system startup. At that time, they are loaded into actual memory and are called MIAP PDTs. The SIOGEN is an independent macro assembly from SYSGEN.

SYNTAX

SIOCON[TROLLER] dname,#chnl [,fdn] [,wdog] [,CL] [,ridmp] [,maxtc] [,ubufs]

dname - is the logical device name of the DCS channel. It must be the same device name as that specified for the corresponding SIODEVICE statement.

chnl - is the channel number. It is a hexadecimal value with the upper byte representing the slot number and the lower byte representing the subchannel. For example, #0306 represents slot 3, subchannel 6. Valid slot numbers are from 0 to #F. Valid channel numbers range from 0 to 7.

[fdn] - (OPTIONAL) is used only for full duplex channels and specifies the logical device name (dname) of the 'mate'. For simplex channels, this parameter must be left blank.

[wdog] - (OPTIONAL) specifies the maximum time, in hundredths of a second, that an operation is allowed to remain at the head of the queue for this channel. The default value is 0, which means that the operation is not timed.

[CL] - (OPTIONAL) is a keyword used to specify that an asynchronous device has a current loop interface.

[ridmp] - is used only for MAXNET channels.

[maxtc] - (OPTIONAL) specifies the maximum transfer count in bytes for REX READ or WRITE operations on this channel. The default is 256 cytes. If present, this parameter must be nonzero. The LIM Monitor uses the largest value of this parameter and parameter 3 (bpr - bytes per record) of the SIODEVICE statement to determine the size of the LIM buffers to allocate for data transfers.

[ubufs] - (OPTIONAL) specifies the number of buffers that should be allocated on the LIM for use by the handler for this subchannel. Most handlers need only 1 (the default). The bitsync handler needs a minimum of 3 (its default).

The sum of the values of this parameter for all the subchannels SYSGENed for a LIM, plus one, makes up the total number of identically sized buffers that the LIM Monitor will allocate on the LIM. How the buffer size is determined is described under parameter 7, maxtc.

If too many large buffers are requested and there is not enough memory on the LIM, the user will get an error from the Loader task indicating that there is insufficent memory on the LIM card. To correct this, reduce the value of this parameter or parameter 7, maxtc, or parameter 3, bpr, of the SIODEVICE statement.

4.5.4 SIODEVICE Statement

The SIODEVICE statement must be specified for each DCS channel to be defined. This statement generates a Logical Device Table (LDT). The LDTs that are generated are resident in MAP 0.

SYNTAX

```
SIODEV[ICE]     dname,cname [,bpr] [,opt] [,xtask]
                                          [,-infl]
```

dname - is the logical device name (1 to 3 CAN code characters) and must be unique among all device names.

cname - is the MUX controller name which was specified as parameter 1 of the associated MUXCONTROLLER statement.

[bpr] - (OPTIONAL) specifies the record size in bytes (bytes per record) for this device. Valid values are 0 through 510. If 0 (the default) is specified, the record size is considered unlimited. See also parameter 7 (maxtc) of the SIOCONTROLLER statement.

[opt] - permits modification of the contents of the logical device characteristics word in the data structure (LDT) for this device. It specifies certain options that may be examined by the specific handler when a handler supports many similar but not identical types of devices. This LDT option word is shown in the specific DCS handler chapters.

[xtask]
[-influ] - (OPTIONAL) specifies either a task name or a negative influence limit. If a task name is given, the designated task is given permanent exclusive use of the device to the complete exclusion of all other tasks.

If a negative influence limit number in the range of -000 or -255 is given, then only tasks that have an influence limit number with absolute value less than or equal to the absolute value of this parameter are permitted to use this logical device. The default is a negative influence limit number of -255.

4.5.5 LINEMONITOR Statement

DEFINE COMMUNICATION LINE MONITORING CHARACTERISTICS

One LINEMONITOR statement is optional and is used only if a ring processing task is to be included in the system. This statement is required for each communications line for which ring is queued to some line monitoring support task. One such task is supplied by MODCOMP to provide for auto answer support (this task is called the Terminal Monitor Program - TMP); however, any number of user tasks may also exist in the system.

SYNTAX

LINEMON output [,input] [,options] [,task] [,prefix] [,delay]

output - is the logical device name on which ring and line disconnect are to be detected. If the device is a full-duplex device, "output" is the name of the device on the output channel.

[input] - is only required if the device is a full-duplex device. If the device is half-duplex, either this parameter must not be entered or must be entered the same as parameter 1. If the device on which ring and line disconnect are to be detected is a full-duplex device, "input" is the logical device name of the device connected to the input channel.

[options] - (OPTIONAL) specifies line monitoring characteristics as follows:

RING	(#1000)	Ring Support
CONN	(#0800)	Auto Task Line Connect Support
DISC	(#0400)	Auto Task Line Disconnect Support
DELAY	(#0200)	Line Disconnect Delay Support

Auto task line connect support, auto task line disconnect support, and line disconnect delay support require that ring support be optioned. The default option is no ring support.

[task] - (OPTIONAL) specifies the task name that is to be notified (by a packet queued to it) when a break, ring, or line disconnect is detected from the supported device. All these task names must be only 3-characters long. The default task name is TMP (Terminal Monitor Program).

[prefix] - (OPTIONAL) specifies a 3-character CAN-codable task name prefix to be used as the prefix to the initial attentive task name. The 3-character suffix to the task name will be the line number in decimal ASCII. A line number is based on the LINEMONITOR statement ordering. The line number of the first LINEMONITO statement is "001". The line number of each additional statement is "002", "003", et cetera. If this parameter is not supplied then there will be no initial attentive task name.

[delay] - (OPTIONAL) specifies the number of ticks to delay after detecting a line disconnection. The time delay operates at low clock level. This parameter is used in conjunction with line disconnect delay support (DELAY) only and the default delay is 3 seconds or 600 ticks.

Programmer's Error Bit In UFT Word UFTSTA	Operator's Reason Code	Reason For Error If Set
0	ER	-Some error exists
1	OVF	- Underrun/Overflow
2	PAR	- CRC error (input)
3	INO	- Power off or watchdog timer expired
4	MPE	- Extended status in UFTBCT, UFTFAT
5	LOK	- Device under exclusive use.
6	CIG	- Transmit abort received.
7	OTH	- Input = CD dropped - Output = CTS dropped

Table 4-1. SDLC/HDLC Frame Handler Errors and Events

4.6 SYSGEN AND SIOGEN EXAMPLE

The following statements are needed in the Logical I/O block of the SYSGEN to configure 4 Bitsync full-duplex channels:

```
MUXCON    DC1,#05,5

SIODEV    S00,DC1,0
SIODEV    S01,DC1,0
```

```
SIODEV          S02,DC1,0
SIODEV          S03,DC1,0
SIODEV          S04,DC1,0
SIODEV          S05,DC1,0
SIODEV          S06,DC1,0
SIODEV          S07,DC1,0
```

The following statements are needed in the SIOGEN corresponding with the above SIODEVICEs:

```
SIOCHARACTERISTIC     B,#05,DLC,,,0

SIOCONTROLLER         S00,#0500,S01,100,CL
SIOCONTROLLER         S01,#0501,S00,100,CL
SIOCONTROLLER         S02,#0502,S03,100,CL
SIOCONTROLLER         S03,#0503,S02,100,CL
SIOCONTROLLER         S04,#0504,S05,100,CL
SIOCONTROLLER         S05,#0505,S04,100,CL
SIOCONTROLLER         S06,#0506,S07,100,CL
SIOCONTROLLER         S07,#0507,S06,100,CL
```

INDEX

Fold

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 3624 FT. LAUDERDALE, FL 33309

POSTAGE WILL BE PAID BY ADDRESSEE

MODULAR COMPUTER SYSTEMS
1650 W. McNAB ROAD
P.O. BOX 6099
FT. LAUDERDALE, FLORIDA 33310

Attention: TECHNICAL PUBLICATIONS, M.S. #85

Fold

MODCOMP

Please comment on the publication's completeness, accuracy, and readability. We also appreciate any general suggestions you may have to improve this publication.

If you found errors in this publication, please specify the page number or include a copy of the page with your remarks.

Your comments will be promptly investigated and appropriate action will be taken. If you require a written answer, please check the box and include your address below.

☐

Comments: ______

Manual Title ______

Manual Order Number ______ Issue Date ______

Name ______ Position ______

Company ______

Address ______

______ Telephone (   ) ______

MODCOMP

Corporate Headquarters:
MODULAR COMPUTER SYSTEMS, Inc., 1650 West McNab Road, P.O. Box 6099, Ft. Lauderdale, FL 33310, Tel (305) 974-1380, TWX 310-372-7837
International Headquarters:
MODULAR COMPUTER SERVICES, Inc., The Business Centre, Molly Millars Lane, Wokingham, Berkshire, RG11 2JQ, UK, Tel 0734-786808, TLX 851849149
Latin American Sales Headquarters
MODULAR COMPUTER SYSTEMS, Inc., 1650 West McNab Road, P.O. Box 6099, Ft. Lauderdale, FL 33310, Tel (305) 975-6562, TLX 3727852
Canadian Headquarters
MODCOMP Canada, Ltd., 400 Matheson Blvd. East, Unit 24, Mississauga, Ontario, Canada L4Z 1N8, Tel (416) 890-0666, TELEX 06-961279

SALES & SERVICE LOCATIONS THROUGHOUT THE WORLD